09/295,690

3/12/02

Refine Search

Search Results -

Terms	Documents
L28 and message	1

Database: US Pre-Grant Publication Full-Text Database; US Patents Full-Text Database; US OCR Full-Text Database; EPO Abstracts Database; JPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Search: L30

Refine Search | Recall Text | Clear | Interrupt

Search History

DATE: Saturday, March 17, 2007 **Purge Queries** Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L30	L28 and message	1	L30
L29	L28 and generat$3	1	L29
L28	(L25 or L26 or L27) and (upgrad$3 with version)	1	L28
L27	william near buckley	76	L27
L26	alex near truong	4	L26
L25	jerome near mouton	2	L25
L24	(L21 or L22 or L23) and (L9 or L10)	8	L24
L23	717/170.ccls.	516	L23
L22	717/122.ccls.	180	L22
L21	715/511.ccls.	470	L21
L20	L15 and (old near release)	2	L20
L19	L15 and (new near release)	17	L19
L18	L17 and (old near version)	14	L18
L17	L15 and (new near version)	36	L17

L16	L15 and (upgrad$3 with version with format$)	1	L16
L15	(L13 or L14) and map$4	57	L15
L14	(L11 or L12) and L10	32	L14
L13	(L11 or L12) and L9	89	L13
L12	717/$.ccls.	13470	L12
L11	707/$.ccls.	41814	L11
L10	L5 and ((recent or latest) with version)	86	L10
L9	L8 and ((recent or latest) with version)	437	L9
L8	L7 and (version with format)	2965	L8
L7	L6 and version	40945	L7
L6	(generat$3 or creat$3 or build$3 or produc$3) with message	140754	L6
L5	L3 and L4	441	L5
L4	message with version	10856	L4
L3	(L1 or L2) and upgrad$3	1845	L3
L2	(updat$3 or chang$3) near message	17203	L2
L1	(intermediate or secondary) near version$	905	L1

END OF SEARCH HISTORY

09/295,690

Refine Search

Search Results -

Terms	Documents
(L21 or L22 or L23) and (L9 or L10)	8

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L24

Refine Search

Recall Text | Clear | Interrupt

Search History

DATE: Saturday, March 17, 2007 **Purge Queries** Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L24	(L21 or L22 or L23) and (L9 or L10)	8	L24
L23	717/170.ccls.	516	L23
L22	717/122.ccls.	180	L22
L21	715/511.ccls.	470	L21
L20	L15 and (old near release)	2	L20
L19	L15 and (new near release)	17	L19
L18	L17 and (old near version)	14	L18
L17	L15 and (new near version)	36	L17
L16	L15 and (upgrad$3 with version with format$)	1	L16
L15	(L13 or L14) and map$4	57	L15
L14	(L11 or L12) and L10	32	L14
L13	(L11 or L12) and L9	89	L13
L12	717/$.ccls.	13470	L12
L11	707/$.ccls.	41814	L11

L10	L5 and ((recent or latest) with version)	86	L10
L9	L8 and ((recent or latest) with version)	437	L9
L8	L7 and (version with format)	2965	L8
L7	L6 and version	40945	L7
L6	(generat$3 or creat$3 or build$3 or produc$3) with message	140754	L6
L5	L3 and L4	441	L5
L4	message with version	10856	L4
L3	(L1 or L2) and upgrad$3	1845	L3
L2	(updat$3 or chang$3) near message	17203	L2
L1	(intermediate or secondary) near version$	905	L1

END OF SEARCH HISTORY

3/17/07

09/295,690

Refine Search

Search Results -

Terms	Documents
L15 and (old near release)	2

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L20

Refine Search

Recall Text Clear Interrupt

Search History

DATE: Saturday, March 17, 2007 **Purge Queries** Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L20	L15 and (old near release)	2	L20
L19	L15 and (new near release)	17	L19
L18	L17 and (old near version)	14	L18
L17	L15 and (new near version)	36	L17
L16	L15 and (upgrad$3 with version with format$)	1	L16
L15	(L13 or L14) and map$4	57	L15
L14	(L11 or L12) and L10	32	L14
L13	(L11 or L12) and L9	89	L13
L12	717/$.ccls.	13470	L12
L11	707/$.ccls.	41814	L11
L10	L5 and ((recent or latest) with version)	86	L10
L9	L8 and ((recent or latest) with version)	437	L9
L8	L7 and (version with format)	2965	L8
L7	L6 and version	40945	L7

L6	(generat$3 or creat$3 or build$3 or produc$3) with message	140754	L6
L5	L3 and L4	441	L5
L4	message with version	10856	L4
L3	(L1 or L2) and upgrad$3	1845	L3
L2	(updat$3 or chang$3) near message	17203	L2
L1	(intermediate or secondary) near version$	905	L1

END OF SEARCH HISTORY

09/295,690

3/12/07

Refine Search

Search Results -

Terms	Documents
L17 and (old near version)	14

Database: US Pre-Grant Publication Full-Text Database; US Patents Full-Text Database; US OCR Full-Text Database; EPO Abstracts Database; JPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Search: L18

Refine Search | Recall Text | Clear | Interrupt

Search History

DATE: Saturday, March 17, 2007 **Purge Queries** Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L18	L17 and (old near version)	14	L18
L17	L15 and (new near version)	36	L17
L16	L15 and (upgrad$3 with version with format$)	1	L16
L15	(L13 or L14) and map$4	57	L15
L14	(L11 or L12) and L10	32	L14
L13	(L11 or L12) and L9	89	L13
L12	717/$.ccls.	13470	L12
L11	707/$.ccls.	41814	L11
L10	L5 and ((recent or latest) with version)	86	L10
L9	L8 and ((recent or latest) with version)	437	L9
L8	L7 and (version with format)	2965	L8
L7	L6 and version	40945	L7
L6	(generat$3 or creat$3 or build$3 or produc$3) with message	140754	L6
L5	L3 and L4	441	L5

L4	message with version	10856	L4
L3	(L1 or L2) and upgrad$3	1845	L3
L2	(updat$3 or chang$3) near message	17203	L2
L1	(intermediate or secondary) near version$	905	L1

END OF SEARCH HISTORY

09/295,690

PORTAL
USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: The ACM Digital Library The Guide

"upgraded version" + "version format generating" + "mapping

SEARCH

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used **upgraded version version format generating mapping final update message**

Found **13** of **198,617**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 13 of 13

Relevance scale

1 Programming the FlexRAM parallel intelligent memory system

Basilio B. Fraguela, Jose Renau, Paul Feautrier, David Padua, Josep Torrellas

June 2003 **ACM SIGPLAN Notices , Proceedings of the ninth ACM SIGPLAN symposium on Principles and practice of parallel programming PPoPP '03**, Volume 38 Issue 10

Publisher: ACM Press

Full text available: pdf(256.04 KB) Additional Information: full citation, abstract, references, citings, index terms

In an intelligent memory architecture, the main memory of a computer is enhanced with many simple processors. The result is a highly-parallel, heterogeneous machine that is able to exploit computation in the main memory. While several instantiations of this architecture have been proposed, the question of how to effectively program them with little effort has remained a major challenge.In this paper, we show how to effectively hand-program an intelligent memory architecture at a high level and w ...

Keywords: compiler directives, intelligent memory architecture, parallel languages, programming heterogeneous computers

2 HydroJ: object-oriented pattern matching for evolvable distributed systems

Keunwoo Lee, Anthony LaMarca, Craig Chambers

October 2003 **ACM SIGPLAN Notices , Proceedings of the 18th annual ACM SIGPLAN conference on Object-oriented programing, systems, languages, and applications OOPSLA '03**, Volume 38 Issue 11

Publisher: ACM Press

Full text available: pdf(311.06 KB) Additional Information: full citation, abstract, references, citings, index terms

In an evolving software system, components must be able to change independently while remaining compatible with their peers. One obstacle to independent evolution is the *brittle parameter problem*: the ability of two components to communicate can depend on a number of *inessential* details of the types, structure, and/or contents of the values communicated. If these details change, then the components can no longer communicate, even if the *essential* parts of the message remain ...

Keywords: HydroJ, XML, distributed systems, dynamic dispatch, object-oriented programming, pattern matching, semi-structured data, software evolution, ubiquitous

computing

3 Cache performance for multimedia applications

Nathan T. Slingerland, Alan Jay Smith

June 2001 **Proceedings of the 15th international conference on Supercomputing ICS '01**

Publisher: ACM Press

Full text available: pdf(642.63 KB) Additional Information: full citation, abstract, references, citings, index terms

The caching behavior of multimedia applications has been described as having high instruction reference locality within small loops, very large working sets, and poor data cache performance due to non-locality of data references. Despite this, there is no published research deriving or measuring these qualities. Utilizing the previously developed Berkeley Multimedia Workload, we present the results of execution driven cache simulations with the goal of aiding future media processing architect ...

Keywords: CPU caches, cache, mulitmedia, simulation, trace driven simulation

4 Distrbuted VEEs: PDS: a virtual execution environment for software deployment

Bowen Alpern, Joshua Auerbach, Vasanth Bala, Thomas Frauenhofer, Todd Mummert, Michael Pigott

June 2005 **Proceedings of the 1st ACM/USENIX international conference on Virtual execution environments VEE '05**

Publisher: ACM Press

Full text available: pdf(299.26 KB) Additional Information: full citation, abstract, references, citings, index terms

The Progressive Deployment System (PDS) is a virtual execution environment and infrastructure designed specifically for deploying software, or "assets", on demand while enabling management from a central location. PDS intercepts a select subset of system calls on the target machine to provide a partial virtualization at the operating system level. This enables an asset's install-time environment to be reproduced virtually while otherwise not isolating the asset from peer applications on the targ ...

Keywords: deployment, installation, management, streaming, virtualization

5 Bridging semantic gap

Pronab Ganguly, Fethi A. Rabhi, Pradeep K. Ray

June 2003 **Proceedings of the 2002 conference on Pattern languages of programs - Volume 13 CRPIT '02**

Publisher: Australian Computer Society, Inc.

Full text available: pdf(122.74 KB) Additional Information: full citation, references, index terms

6 Aspect reuse and domain-specific approaches: Doxpects: aspects supporting XML transformation interfaces

Eric Wohlstadter, Kris De Volder

March 2006 **Proceedings of the 5th international conference on Aspect-oriented software development AOSD '06**

Publisher: ACM Press

Full text available: pdf(149.44 KB) Additional Information: full citation, abstract, references

In the web services environment software development can involve writing both object-oriented programs and XML transformations. This can be seen in the popular Web Services architecture. In this architecture, crosscutting concerns are often manifest as transformations on XML messages; encrypting messages, adapting between schemas types or adding extra-functional elements such as transaction contexts can be seen as prime examples. Some existing middleware platforms provide support for *Handlers* ...

7 Isolation-only transactions by typing and versioning

Paweł T. Wojciechowski

July 2005 **Proceedings of the 7th ACM SIGPLAN international conference on Principles and practice of declarative programming PPDP '05**

Publisher: ACM Press

Full text available: pdf(248.04 KB) Additional Information: full citation, abstract, references, index terms

In this paper we design a language and runtime support for isolation-only, multithreaded transactions (called tasks). Tasks allow isolation to be *declared* instead of having to be encoded using the low-level synchronization constructs. The key concept of our design is the use of a type system to support rollback-free and safe runtime execution of tasks.We present a first-order type system which can verify information for the concurrency controller. We use an operational semantics to formal ...

Keywords: abstract types, concurrency, declarative synchronization, isolation, lambda calculus, programming languages, singleton kinds, transactions, type theory

8 AIDA: Adaptive application-independent data aggregation in wireless sensor networks

Tian He, Brian M. Blum, John A. Stankovic, Tarek Abdelzaher

May 2004 **ACM Transactions on Embedded Computing Systems (TECS)**, Volume 3 Issue 2

Publisher: ACM Press

Full text available: pdf(1.02 MB) Additional Information: full citation, abstract, references, citings, index terms

Sensor networks, a novel paradigm in distributed wireless communication technology, have been proposed for various applications including military surveillance and environmental monitoring. These systems deploy heterogeneous collections of sensors capable of observing and reporting on various dynamic properties of their surroundings in a time sensitive manner. Such systems suffer bandwidth, energy, and throughput constraints that limit the quantity of information transferred from end-to-end. The ...

Keywords: Data aggregation, adaptive algorithms, congestion control, energy conservation, feedback control, sensor networks

9 Optical link and processor clustering in the delft parallel processor

L. Dekker, E. E. E. Frietman

June 1988 **Proceedings of the 2nd international conference on Supercomputing ICS '88**

Publisher: ACM Press

Full text available: pdf(1.91 MB) Additional Information: full citation, abstract, references, index terms

The avoidance of transfer-bound processing in large parallel computers is first analyzed with respect to the processor interconnectability. A quantitative measure is introduced for the processor substask interactivity, based on the number on nonzeroes in the interaction matrix. A similar quantitative measure can be introduced for the processor interconnectability in case of a multi-bus communication system. For a parallel computer with p processors and p busses the asymptotic speedup for la ...

10 HPF
Mike Delves
January 1998 **Linux Journal**
Publisher: Specialized Systems Consultants, Inc.
Full text available: html(10.40 KB) Additional Information: full citation, abstract, references, index terms

Programming Linux Clusters the Easy Way: Mr. Delves tells us all about high performance Fortran and how it is used to write code to run efficiently on parallel computers

11 Directory of educational discounts, donations, and grants
Hubert Callihan
August 1992 **ACM SIGGRAPH Computer Graphics**, Volume 26 Issue 3
Publisher: ACM Press
Full text available: pdf(454.41 KB) Additional Information: full citation, abstract, index terms

The information included here was compiled using responses from 36 hardware and software vendors who were among 189 exhibitors attending the 1991 SIGGRAPH Conference in Las Vegas. Product descriptions and other details vary, but in most cases, provide enough information for educators to inquire further about opportunities to obtain products at a discount, as a donation, or possibly as a grant in connection with academic research. Companies responding that have neither educational discount, donat ...

12 Poster session: papers included: Optimized concrete delivery scheduling using combined simulation and genetic algorithms
Ming Lu, Hoi-Ching Lam
December 2005 **Proceedings of the 37th conference on Winter simulation WSC '05**
Publisher: Winter Simulation Conference
Full text available: pdf(493.91 KB) Additional Information: full citation, abstract, references

The research presented is mainly focused on how to simultaneously optimize concrete delivery scheduling and resource provisions for ready mixed concrete (RMC) plants based on a valid simulation modeling platform resulting from research, called HKCONSIM. Combined discrete-event simulation and genetic algorithms (GA) are applied in HKCONSIM to model and further optimize the one-plant-multisite RMC plant operations in Hong Kong. Logistics planning practices of RMC businesses in Hong Kong are introd ...

13 Improving the adaptability of multi-mode systems via program steering
Lee Lin, Michael D. Ernst
July 2004 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 2004 ACM SIGSOFT international symposium on Software testing and analysis ISSTA '04**, Volume 29 Issue 4
Publisher: ACM Press
Full text available: pdf(195.32 KB) Additional Information: full citation, abstract, references, citings, index terms

A multi-mode software system contains several distinct modes of operation and a controller for deciding when to switch between modes. Even when developers rigorously test a multi-mode system before deployment, they cannot foresee and test for every possible usage scenario. As a result, unexpected situations in which the program fails or underperforms (for example, by choosing a non-optimal mode) may arise. This research aims to mitigate such problems by creating a new mode selector that examines ...

Keywords: adaptability, mode selection, multi-mode systems, program steering

Results 1 - 13 of 13

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2007 ACM, Inc.
Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player

09/295,690

3/17/07

PORTAL
USPTO
Subscribe (Full Service) Register (Limited Service, Free) Login
Search: The ACM Digital Library The Guide
"upgraded version chaining" + "repeatedly generating" + "mapp
SEARCH
THE ACM DIGITAL LIBRARY
Feedback Report a problem Satisfaction survey

Terms used **upgraded version chaining repeatedly generating mapping last update message**

Found **34** of **198,617**

Sort results by: relevance
Display results: expanded form
Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 34 Result page: **1** 2 next

Relevance scale

1 A probabilistic language based upon sampling functions

Sungwoo Park, Frank Pfenning, Sebastian Thrun
January 2005 **ACM SIGPLAN Notices , Proceedings of the 32nd ACM SIGPLAN-SIGACT symposium on Principles of programming languages POPL '05**, Volume 40 Issue 1
Publisher: ACM Press
Full text available: pdf(282.09 KB) Additional Information: full citation, abstract, references, index terms

As probabilistic computations play an increasing role in solving various problems, researchers have designed probabilistic languages that treat probability distributions as primitive datatypes. Most probabilistic languages, however, focus only on discrete distributions and have limited expressive power. In this paper, we present a probabilistic language, called λo, which uniformly supports all kinds of probability distributions -- discrete distributions, continuous distributions, and e ...

Keywords: probabilistic language, probability distribution, robotics, sampling function

2 Parallelism in logic programs

Raghu Ramakrishnan
December 1989 **Proceedings of the 17th ACM SIGPLAN-SIGACT symposium on Principles of programming languages POPL '90**
Publisher: ACM Press
Full text available: pdf(1.76 MB) Additional Information: full citation, references, citings, index terms

3 Behavior models: Mining object behavior with ADABU

Valentin Dallmeier, Christian Lindig, Andrzej Wasylkowski, Andreas Zeller
May 2006 **Proceedings of the 2006 international workshop on Dynamic systems analysis WODA '06**
Publisher: ACM Press
Full text available: pdf(142.63 KB) Additional Information: full citation, abstract, references, index terms

To learn what constitutes *correct* program behavior, one can start with *normal* behavior. We observe actual program executions to construct *state machines* that summarize object behavior. These state machines, called *object behavior models*, capture the relationships between two kinds of methods: *mutators* that change the state (such as add()) and

inspectors that keep the state unchanged (such as isEmpty()): "A Vector object initially is in isEmpty() state; after ad ...

Keywords: automata, inspector method, java, mining, object behavior model, object state, observer method, program analysis, program instrumentation, purity analysis

4 Evolutionary combinatorial optimization: The blob code is competitive with edge-sets in genetic algorithms for the minimum routing cost spanning tree problem

Bryant A. Julstrom

June 2005 **Proceedings of the 2005 conference on Genetic and evolutionary computation GECCO '05**

Publisher: ACM Press

Full text available: pdf(109.33 KB) Additional Information: full citation, abstract, references, index terms

Among the many codings of spanning trees for evolutionary search are those based on bijections between Prüfer strings---strings of n-2 vertex labels---and spanning trees on the labeled vertices. One of these bijections, called the Blob Code, showed promise as an evolutionary coding, but EAs that use it to represent spanning trees have not performed well. Here, a genetic algorithm that represents spanning trees via the Blob Code is faster than, and returns results competitive with tho ...

Keywords: blob code, codings, edge-sets, routing cost, spanning trees

5 Composing distributed applications through generative communication

Tom Holvoet, Yolande Berbers

September 1998 **Proceedings of the 8th ACM SIGOPS European workshop on Support for composing distributed applications EW 8**

Publisher: ACM Press

Full text available: pdf(652.82 KB) Additional Information: full citation, index terms

6 A product-line requirements approach to safe reuse in multi-agent systems

Josh Dehlinger, Robyn R. Lutz

May 2005 **ACM SIGSOFT Software Engineering Notes , Proceedings of the fourth international workshop on Software engineering for large-scale multi-agent systems SELMAS '05**, Volume 30 Issue 4

Publisher: ACM Press

Full text available: pdf(84.47 KB) Additional Information: full citation, abstract, references, index terms

The dynamic nature of highly autonomous agents within distributed systems is difficult to specify with existing requirements techniques. However, capturing the possibly shifting configurations of agents in the requirements specification is essential for safe reuse of agents. The contribution of this work is an extensible agent-oriented requirements specification template for distributed systems that supports safe reuse. We make two basic claims for this idea. First, by adopting a product-line-li ...

7 Recent technical reports

January 1979 **ACM SIGACT News**, Volume 10 Issue 3

Publisher: ACM Press

Full text available: pdf(1.99 MB) Additional Information: full citation

8

Optimising Web queries using document type definitions
Peter T. Wood
November 1999 **Proceedings of the 2nd international workshop on Web information and data management WIDM '99**
Publisher: ACM Press
Full text available: pdf(602.00 KB) Additional Information: full citation, abstract, references, citings, index terms

A document type definition (DTD) D defines the structure of elements permitted in any web document valid with respect to D. From a given DTD D we show how to derive a number of simple structural constraints which are implied by D. Using a relational abstraction of web databases, we consider a class of conjunctive queries which retrieve elements from web documents stored in a database D. For simplicity, we a ...

9 Animating and rendering liquids: Practical animation of turbulent splashing water
Janghee Kim, Deukhyun Cha, Byungjoon Chang, Bonki Koo, Insung Ihm
September 2006 **Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation SCA '06**
Publisher: Eurographics Association
Full text available: pdf(601.69 KB) Additional Information: full citation, abstract, references, index terms

Despite recent advances in fluid animation, producing small-scale detail of turbulent water still remains challenging. In this paper, we extend the well-accepted particle level set method in an attempt to integrate the dynamic behavior of splashing water easily into a fluid animation system. Massless marker particles that still escape from the main body of water, in spite of the level set correction, are transformed into water particles to represent subcell-level features that are hard to captur ...

10 Exploiting syntactic structure for language modeling
Ciprian Chelba, Frederick Jelinek
August 1998 **Proceedings of the 36th annual meeting on Association for Computational Linguistics - Volume 1 , Proceedings of the 17th international conference on Computational linguistics - Volume 1**
Publisher: Association for Computational Linguistics
Full text available: pdf(608.59 KB) Publisher Site Additional Information: full citation, abstract, references, citings

The paper presents a language model that develops syntactic structure and uses it to extract meaningful information from the word history, thus enabling the use of long distance dependencies. The model assigns probability to every joint sequence of words-binary-parse-structure with headword annotation and operates in a left-to-right manner --- therefore usable for automatic speech recognition. The model, its probabilistic parameterization, and a set of experiments meant to evaluate its predictiv ...

11 Gene selection criterion for discriminant microarray data analysis based on extreme value distributions
Wentian Li, Ivo Grosse
April 2003 **Proceedings of the seventh annual international conference on Research in computational molecular biology RECOMB '03**
Publisher: ACM Press
Full text available: pdf(158.65 KB) Additional Information: full citation, abstract, references, citings, index terms

An important issue commonly encountered in the analysis of microarray data is to decide which and how many genes should be selected for further studies. For discriminant microarray data analyses based on statistical models, such as the logistic regression model, this gene selection can be accomplished by a comparison of the maximum

likelihood of the model given the real data, L(D|M), and the expected maximum likelihood of the model given an ensemble of surrogate data, L(D0|M). Typical ...

Keywords: classification, extreme values, logistic regression, microarray

12 Live documents with contextual, data-driven information components

Anke Weber, Holger M. Kienle, Hausi A. Müller

October 2002 **Proceedings of the 20th annual international conference on Computer documentation SIGDOC '02**

Publisher: ACM Press

Full text available: pdf(627.10 KB) Additional Information: full citation, abstract, references, citings, index terms

We introduce the notion of a live document and we describe our concept of live documents with contextual, data driven information components. The dynamic and interactive features of live documents provide a consistent data source for multimedia presentations targeted to various audiences and multiple platforms. Therefore, they contribute to the solution of key challenges in single sourcing and repurposing. We motivate the use of live documents with sample scenarios from the field of systems docu ...

Keywords: Microsoft Office, live documents, repurposing, reverse engineering, scalable vector graphics, single sourcing, software engineering, systems documentation

13 LeZi-update: an information-theoretic approach to track mobile users in PCS networks

Amiya Bhattacharya, Sajal K. Das

August 1999 **Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking MobiCom '99**

Publisher: ACM Press

Full text available: pdf(1.59 MB) Additional Information: full citation, references, citings, index terms

14 Oracles and queries that are sufficient for exact learning (extended abstract)

Nader H. Bshouty, Richard Cleve, Sampath Kannan, Christino Tamon

July 1994 **Proceedings of the seventh annual conference on Computational learning theory COLT '94**

Publisher: ACM Press

Full text available: pdf(951.23 KB) Additional Information: full citation, abstract, references, citings, index terms

We show that the class of all circuits is exactly learnable in randomized expected polynomial-time using subset and superset queries. This is a consequence of the following result which we consider to be of independent interest: circuits are exactly learnable in randomized expected polynomial-time with equivalence queries and the aid of an NP-oracle. We also show that circuits are exactly learnable in deterministic polynomial-time with equivalence queries and a &Sgr;

15 Research track posters: 2PXMiner: an efficient two pass mining of frequent XML query patterns

Liang Huai Yang, Mong Li Lee, Wynne Hsu, Xinyu Guo

August 2004 **Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining KDD '04**

Publisher: ACM Press

Additional Information:

Full text available: pdf(200.65 KB) full citation, abstract, references, citings, index terms

Caching the results of frequent query patterns can improve the performance of query evaluation. This paper describes a 2-pass mining algorithm called 2PXMiner to discover frequent XML query patterns. We design 3 data structures to expedite the mining process. Experiments results indicate that 2PXMiner is both efficient and scalable.

Keywords: XML query pattern, tree mining

16 Research sessions: non-standard query processing: Buffering databse operations for enhanced instruction cache performance

Jingren Zhou, Kenneth A. Ross

June 2004 **Proceedings of the 2004 ACM SIGMOD international conference on Management of data SIGMOD '04**

Publisher: ACM Press

Full text available: pdf(188.52 KB) Additional Information: full citation, abstract, references, citings

As more and more query processing work can be done in main memory access is becoming a significant cost component of database operations. Recent database research has shown that most of the memory stalls are due to second-level cache data misses and first-level instruction cache misses. While a lot of research has focused on reducing the data cache misses, relatively little research has been done on improving the instruction cache performance of database systems.We first answer the question "Why ...

17 Test: Automatic generation of test sets for SBST of microprocessor IP cores

E. Sanchez, M. Reorda Reorda, G. Squillero, M. Violante

September 2005 **Proceedings of the 18th annual symposium on Integrated circuits and system design SBCCI '05**

Publisher: ACM Press

Full text available: pdf(258.50 KB) Additional Information: full citation, abstract, references, index terms

Higher integration densities, smaller feature lengths, and other technology advances, as well as architectural evolution, have made microprocessor cores exceptionally complex. Currently, Software-Based Self-Test (SBST) is becoming an attractive test solution since it guarantees high fault coverage figures, runs at-speed, and matches core test requirements while exploiting low-cost ATEs. However, automatically generating test programs is still an open problem. This paper presents a novel approach ...

Keywords: FPGA, automatic test generation, hardware accelerator, microprocessor test, pipelined architectures, test programs

18 Advances in dataflow programming languages

Wesley M. Johnston, J. R. Paul Hanna, Richard J. Millar

March 2004 **ACM Computing Surveys (CSUR)**, Volume 36 Issue 1

Publisher: ACM Press

Full text available: pdf(835.52 KB) Additional Information: full citation, abstract, references, citings, index terms

Many developments have taken place within dataflow programming languages in the past decade. In particular, there has been a great deal of activity and advancement in the field of dataflow visual programming languages. The motivation for this article is to review the content of these recent developments and how they came about. It is supported by an initial review of dataflow programming in the 1970s and 1980s that led to current topics of research. It then discusses how dataflow programming evo ...

Keywords: Dataflow, co-ordination languages, component software, data flow visual programming, graphical programming, multithreading, software engineering

19 Computational structure of a performance assessment involving stochastic and subjective uncertainty

Jon C. Helton

November 1996 **Proceedings of the 28th conference on Winter simulation WSC '96**

Publisher: ACM Press, IEEE Computer Society

Full text available: pdf(803.01 KB) Additional Information: full citation, abstract, references, citings

A recent performance assessment for the Waste Isolation Pilot Plant (WIPP), which is being developed by the U.S. Department of Energy for the geologic disposal of transuranic waste, is used to illustrate the computational structure of a large analysis that maintains a separation between stochastic (i.e., aleatory) and subjective (i.e., epistemic) uncertainty. In this analysis, stochastic uncertainty arises from the many possible disruptions that could occur over the 10,000 yr regulatory period t ...

20 Modeling and visualization of leaf venation patterns

Adam Runions, Martin Fuhrer, Brendan Lane, Pavol Federl, Anne-Gaëlle Rolland-Lagan, Przemyslaw Prusinkiewicz

July 2005 **ACM Transactions on Graphics (TOG) , ACM SIGGRAPH 2005 Papers SIGGRAPH '05**, Volume 24 Issue 3

Publisher: ACM Press

Full text available: pdf(1.05 MB) mov(22:51 MIN) Additional Information: full citation, abstract, references, index terms

We introduce a class of biologically-motivated algorithms for generating leaf venation patterns. These algorithms simulate the interplay between three processes: (1) development of veins towards hormone (auxin) sources embedded in the leaf blade; (2) modification of the hormone source distribution by the proximity of veins; and (3) modification of both the vein pattern and source distribution by leaf growth. These processes are formulated in terms of iterative geometric operations on sets of poi ...

Keywords: Voronoi diagram, auxin, leaf growth, modeling of natural phenomena, morphogenesis, realistic image synthesis, relative neighborhood, vein development

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2007 ACM, Inc.
Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player

09/295,690

IEEE Xplore® RELEASE 2.3

Welcome United States Patent and Trademark Office

Results for "(((upgrad* <paragraph> version) <and> format*)<in>metadata)"

e-mail

Your search matched **2** of **1520067** documents.

A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

Modify Search

(((upgrad* <paragraph> version) <and> format*)<in>metadata) Search

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IET JNL	IET Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IET CNF	IET Conference Proceeding
IEEE STD	IEEE Standard

view selected items Select All Deselect All

1. **Safe upgrading without restarting**
Barr, M.; Eisenbach, S.;
Software Maintenance, 2003. ICSM 2003. Proceedings. International Conferer
22-26 Sept. 2003 Page(s):129 - 137
Digital Object Identifier 10.1109/ICSM.2003.1235414
AbstractPlus | Full Text: PDF(511 KB) **IEEE CNF**
Rights and Permissions

2. **Current status of crossed-field devices**
MacMaster, G.H.;
Electron Devices Meeting, 1988. Technical Digest., International
11-14 Dec. 1988 Page(s):358 - 361
Digital Object Identifier 10.1109/IEDM.1988.32830
AbstractPlus | Full Text: PDF(496 KB) **IEEE CNF**
Rights and Permissions

Indexed by Inspec®

© Copyright 2006 IEEE –

09/295,690

3/17/07

IEEE Xplore® RELEASE 2.3

Welcome United States Patent and Trademark Office

Results for "(((upgrad* <paragraph> version) <and> map*)<in>metadata)" e-mail
Your search matched **6** of **1520067** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

Modify Search

(((upgrad* <paragraph> version) <and> map*)<in>metadata) Search

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IET JNL	IET Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IET CNF	IET Conference Proceeding
IEEE STD	IEEE Standard

view selected items Select All Deselect All

1. **Operational method to correct VEGETATION satellite measurements from effects**
Berthelot, B.; Dedieu, G.;
Geoscience and Remote Sensing Symposium, 2000. Proceedings. IGARSS 2(International
Volume 2, 24-28 July 2000 Page(s):831 - 833 vol.2
Digital Object Identifier 10.1109/IGARSS.2000.861718
AbstractPlus | Full Text: PDF(244 KB) **IEEE CNF**
Rights and Permissions

2. **The impact of IT on education**
Beckett, J.;
Engineering Science and Education Journal
Volume 5, Issue 4, Aug. 1996 Page(s):185 - 189
AbstractPlus | Full Text: PDF(492 KB) **IET JNL**

3. **The case for run-time replaceable kernel modules**
Draves, R.;
Workstation Operating Systems, 1993. Proceedings., Fourth Workshop on
14-15 Oct. 1993 Page(s):160 - 164
Digital Object Identifier 10.1109/WWOS.1993.348155
AbstractPlus | Full Text: PDF(340 KB) **IEEE CNF**
Rights and Permissions

4. **Enabling technologies to map precipitation with near-global coverage an revisit times**
Ruf, C.S.; Principe, C.M.; Neeck, S.P.;
Geoscience and Remote Sensing Symposium, 2000. Proceedings. IGARSS 2(International
Volume 7, 24-28 July 2000 Page(s):2988 - 2990 vol.7
Digital Object Identifier 10.1109/IGARSS.2000.860311
AbstractPlus | Full Text: PDF(204 KB) **IEEE CNF**
Rights and Permissions

5. **SIR-B-The Second Shuttle Imaging Radar Experiment**
Cimino, J.; Elachi, C.; Settle, M.;
Geoscience and Remote Sensing, IEEE Transactions on
Volume GE-24, Issue 4, July 1986 Page(s):445 - 452

Digital Object Identifier 10.1109/TGRS.1986.289658

AbstractPlus | Full Text: PDF(5622 KB) **IEEE JNL**
Rights and Permissions

6. **A Middleware Framework for Integrating Intelligent Building Systems Bas system Peer Mode**
Zeng Ming; Chen Li-ding; Xu Bu-gong;
Computational Engineering in Systems Applications, IMACS Multiconference c
Oct. 2006 Page(s):1766 - 1770
Digital Object Identifier 10.1109/CESA.2006.313599
AbstractPlus | Full Text: PDF(5117 KB) **IEEE CNF**
Rights and Permissions

Help Contact Us Privacy &
Indexed by Inspec®
© Copyright 2006 IEEE –

09/295,690

Results for "((final <near> updat* <near> message)<in>metadata)" e-mail
Your search matched **2** of **1520067** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

Modify Search

((final <near> updat* <near> message)<in>metadata) Search

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IET JNL	IET Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IET CNF	IET Conference Proceeding
IEEE STD	IEEE Standard

view selected items Select All Deselect All

1. **The UK demonstration of TMC - the final results and update on progress**
Burton, P.; Eveleigh, H.;
Road Transport Information and Control, 2002. Eleventh International Confere
Publ. No. 486)
19-21 March 2002 Page(s):9 - Paul Burton, Helen Eveleigh, O. Farber
AbstractPlus | Full Text: PDF(471 KB) **IET CNF**

2. **Development of a class of distributed termination detection algorithms**
Kumar, D.;
Knowledge and Data Engineering, IEEE Transactions on
Volume 4, Issue 2, April 1992 Page(s):145 - 155
Digital Object Identifier 10.1109/69.134251
AbstractPlus | Full Text: PDF(1068 KB) **IEEE JNL**
Rights and Permissions

Indexed by Inspec

© Copyright 2006 IEEE –

09/295,690

IEEE Xplore® RELEASE 2.3

Welcome United States Patent and Trademark Office

Results for **"(('final update message')<in>metadata)"**

e-mail

Your search matched **0** documents.

A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

Modify Search

(('final update message')<in>metadata) Search

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IET JNL	IET Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IET CNF	IET Conference Proceeding
IEEE STD	IEEE Standard

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistan search.

© Copyright 2006 IEEE –

Indexed by Inspec®

```
File   9:Business & Industry(R) Jul/1994-2007/Mar 06
         (c) 2007  The Gale Group
File  16:Gale Group PROMT(R) 1990-2007/Mar 06
         (c) 2007 The Gale Group
File  47:Gale Group Magazine DB(TM) 1959-2007/Feb 26
         (c) 2007 The Gale group
File 148:Gale Group Trade & Industry DB 1976-2007/Feb 26
         (c)2007 The Gale Group
File 160:Gale Group PROMT(R) 1972-1989
         (c) 1999 The Gale Group
File 275:Gale Group Computer DB(TM) 1983-2007/Mar 06
         (c) 2007 The Gale Group
File 621:Gale Group New Prod.Annou.(R) 1985-2007/Feb 26
         (c) 2007 The Gale Group
File 649:Gale Group Newswire ASAP(TM) 2007/Feb 09
         (c) 2007 The Gale Group
File 636:Gale Group Newsletter DB(TM) 1987-2007/Mar 06
         (c) 2007 The Gale Group
File 647:CMP  Computer Fulltext 1988-2007/May W3
         (c) 2007 CMP Media, LLC
File 674:Computer News Fulltext 1989-2006/Sep W1
         (c) 2006 IDG Communications

Set     Items   Description
S1    4896135   MESSAGE? ? OR DOCUMENT? ? OR FILE OR FILES OR DATAFILE? OR
             SCHEMA? ? OR SCHEMATA? ? OR DATASTRUCTURE?
S2        180   TEXTMESSAGE? OR TEXTFILE? OR TEXTDATA OR COMPUTERFILE?
S3    3062012   OBJECT? ? OR CONTENT? ?
S4   17203480   APPLICATION? ? OR PROGRAM? ? OR PROGRAMME OR PROGRAMMES OR
           APP OR APPS OR SOFTWARE? OR SOFT()WARE? ?
S5        377   SOURCECOD? OR PROGRAMCOD? OR PROGRAMMECOD?
S6     394529   PACKET? ? OR DATAPACKET? OR DATAGRAM? OR DATA()GRAM? ?
S7   12576556   UPDAT???? OR UPGRAD? OR UP()(GRAD???? OR DAT????) OR REVIS-
             ???? OR CHANG??? OR MODIFY? OR MODIFIE? ? OR MODIFICAT?
S8    1640182   ADAPT??? OR ADAPTATION? OR RECONFIGUR? OR RE()CONFIGUR????-
             ?? OR ALTERR? OR ALTERATION? OR ALTER?? OR ALTERING
S9     199180   REFORMAT? OR REFRESH??? OR RE()(FORMAT???? OR FRESH???)
S10   1415640   S7:S9(5N)(S1:S6 OR DATA OR DATUM? ?)
S11   2259625   CHAIN???? OR INTERMEDIATE?
S12    779829   MAP OR MAPS OR MAPPED OR MAPPING? ? OR MAPED OR MAPING? ?
S13   1249873   ITERAT???? OR PERMUTAT? OR EVOLV??? OR EVOLUTION?
S14   9030739   VERSION??? OR ADAPTATION? ? OR REVISION? ? OR EDITION? ? OR
            RELEASE OR RELEASES
S15    316877   S14(5N)(GENERAT???? OR BUILD??? OR BUILT OR CONSTRUCT???? -
           OR CREAT???? OR PRODUCE? ? OR PRODUCING OR PRODUCTION? ?)
S16    378595   S14(5N)(PROD? ? OR DERIV??? OR DERIVATION? ? OR PROPAGAT??-
             ?? OR DEFIN??? OR DEFINITION? OR ESTABLISH? OR DEVELOP?????)
S17     50355   S14(5N)(FORM OR FORMS OR FORMED OR FORMING OR FORMATION? ?)
S18     13599   S15:S17(S)(S11 OR S13)
S19       780   S18(S)S10
S20     12607   S15:S16(S)(S11 OR S13)
S21       756   S20(S)S10
S22       658   S20(S)(S7:S9(5N)S1:S6)
S23       411   S22/1999:2007
S24       247   S22 NOT S23
S25       151   RD  (unique items)
S26        47   S19(S)(FORMAT?? OR FORMATT? OR FORMATING)
S27        12   S26/1999:2007
S28        35   S26 NOT S27
S29        24   RD  (unique items)
S30       134   S25 NOT (S26 OR FATTY)
S31       125   S30 NOT (SUPPLY OR NEWSPAPER)(1W)CHAIN? ?
S32       124   S31 NOT DARWIN
? t32/3,k/35,75,85,87-88,90,95,117
```

32/3,K/35 (Item 2 from file: 47)
DIALOG(R)File 47:Gale Group Magazine DB(TM)
(c) 2007 The Gale group. All rts. reserv.

04689446 SUPPLIER NUMBER: 19125093 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Version-control software. (overview of three evaluations of three version-control software packages) (individual evaluation records searchable under "Version-control software") (Software Review)(Evaluation)
Gagnon, Gabrielle
PC Magazine, v16, n5, p219(3)
March 4, 1997
DOCUMENT TYPE: Evaluation ISSN: 0888-8507 LANGUAGE: English
RECORD TYPE: Fulltext; Abstract
WORD COUNT: 721 LINE COUNT: 00066

... might cost a company millions of dollars. There has to be some way of handling change effectively.
Enter Software Change Management (SCM) tools. SCM products address the problems of administering development projects and avoiding the...

...inherent in team development and ongoing software maintenance. They include facilities for document management and version control, defect tracking, team communications, build management, and reporting.
Version -control products in particular focus on tracking and coordinating changes to documents . They include document management resources to organize files and manage storage, library check-in/check-out and file...

...schemes to identify milestones and group-associated documents, version history to trace each document's evolution , and audit logs to track processing.
Some of the issues peculiar to version control have...

32/3,K/75 (Item 2 from file: 160)
DIALOG(R)File 160:Gale Group PROMT(R)
(c) 1999 The Gale Group. All rts. reserv.

02196395
Softool Corp. releases Change and Configuration Control for HP 9000
News Release April 12, 1989 p. 1

Softool Corp. has announced that its Change and Configuration Control (CCC (R)) software product is now available for the Hewlett- Packard HP (TM) 9000 (Series 300 and 800...

... g., code in any language, objects, executables, command procedures), providing total control; allows only authorized changes to be incorporated into the software , ensuring product integrity; maintains a complete audit trail of changes, permitting complete status accounting and re- creation of prior versions ; manages change activities taking place on different versions of the same information, increasing efficiency and coordination; merges parallel configurations, allowing changes to be easily applied from one evolving version to another, and controls and reports dependencies between components, ensuring that all parts of...

32/3,K/85 (Item 8 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2007 The Gale Group. All rts. reserv.

01765510 SUPPLIER NUMBER: 16704870 (USE FORMAT 7 OR 9 FOR FULL TEXT)

Cross-platform and OOP converge: VisualWorks.(ParcPlace Systems' VisualWorks 2.0 Smalltalk-based application development environment) (Software Review)(Evaluation)
Aquino, Alexander A.
Data Based Advisor, v13, n2, p76(3)
Feb, 1995
DOCUMENT TYPE: Evaluation ISSN: 0740-5200 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 2527 LINE COUNT: 00199

... an object in the application (figure 2).
This link provides a layer that insulates the application from certain changes in the database schema , and vice versa. For example, changes in the spelling of column names don't have to cause changes in several application objects . Only the Mapping Tool is used to update the definition of the object . While this additional layer imposes its own maintenance and runtime overhead, it does make the application more flexible and receptive to iterative changes as an application progresses from a prototype to a production release .
As of this writing, VisualWorks only connects to Oracle 6 and 7 and Sybase databases...

32/3,K/87 (Item 10 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2007 The Gale Group. All rts. reserv.

01697677 SUPPLIER NUMBER: 16197462 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Investigating the hybrid windowing and messaging architecture of Chicago. (Microsoft's Chicago operating system) (Technical)
Pietrek, Matt
Microsoft Systems Journal, v9, n9, p15(14)
Sept, 1994
DOCUMENT TYPE: Technical ISSN: 0889-9932 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 12182 LINE COUNT: 00926

... and system events, the Windows 3.1 TOOLHELP.DLL also provided a convenient way to iterate through various system data structures such as modules, tasks, and heaps. In Chicago, these data structures have changed significantly for 32-bit programs . To its credit, Microsoft implemented a 32-bit version of ToolHelp whose functions are defined in TLHELP32.H. A word of caution however: 32-bit ToolHelp APIs are not supported...

32/3,K/88 (Item 11 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2007 The Gale Group. All rts. reserv.

01685473 SUPPLIER NUMBER: 15387878 (USE FORMAT 7 OR 9 FOR FULL TEXT)
There's no delta. (Omega Systems' Versions version control software for Windows applications development) (Software Review) (Software Tools & Techniques) (Evaluation)
Harris, Philip
EXE, v8, n11, p54(4)
May, 1994
DOCUMENT TYPE: Evaluation ISSN: 0268-6872 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 2911 LINE COUNT: 00216

... source code and the chaos that causes.
Familiar problem
Even with only one programmer, minor changes to source files can introduce obscure new bugs that often require hours of careful debugging to

find. Keeping...

...can help ease the burden by allowing comparisons to find exactly what areas of the program have been changed . But as software packages evolve and new versions are released, managing the backlog of source files becomes almost as time consuming and error prone as the development process itself.
This is where version control systems come in. Under a version control system source files for a project are...

32/3,K/90 (Item 13 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2007 The Gale Group. All rts. reserv.

01669219 SUPPLIER NUMBER: 15062132 (USE FORMAT 7 OR 9 FOR FULL TEXT)
New GUI and design environment for Objectivity/DB. (Classify/DB graphical user interface generator from Micram for the Objectivity/DB data base application development software) (Brief Article) (Product Announcement)
Kelly, Sean; Thorne, Marjorie
DBMS, v7, n2, p38(1)
Feb, 1994
DOCUMENT TYPE: Product Announcement ISSN: 1041-5173 LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT
WORD COUNT: 218 LINE COUNT: 00019

... appropriate GUI builders. Changes to the generated code can be reimported to the repository. All changes are preserved if database schema evolution is applied and new versions of the GUI are generated .
Classify/DB costs approximately $4000 per user. Contact Objectivity Inc., 800 El Camino Real, Menlo...

32/3,K/95 (Item 18 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2007 The Gale Group. All rts. reserv.

01541262 SUPPLIER NUMBER: 12749869 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Big brother. (Software Review) (VCS)(overview of three evaluations of version control systems) (Evaluation)
Saran, Cliff
EXE, v7, n5, p16(4)
Oct, 1992
DOCUMENT TYPE: Evaluation ISSN: 0268-6872 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 1078 LINE COUNT: 00081

... and is stored as a 'revision' (actually revision 1.0). When a programmer wishes to modify the file he 'retrieves' it; makes appropriate changes ; and puts it back when he has finished. This time the VC system will check if there are any differences between the revision stored in the archive (revision 1.0) and the current file being submitted by Put. If there are, the VC system will only store the differences...

...Put cycle will continue under version control. Each iteration of this cycle results in the revision number being incremented.
During the development process there may be times when a development path must be split - for instance to...

32/3,K/117 (Item 11 from file: 636)
DIALOG(R)File 636:Gale Group Newsletter DB(TM)

(c) 2007 The Gale Group. All rts. reserv.

01105353 Supplier Number: 40791921 (USE FORMAT 7 FOR FULLTEXT)
CHANGE & CONFIGURATION CONTROL FROM SOFTOOL NOW ON HEWLETT-PACKARD 9000s
Computergram International, n1176, pN/A
May 15, 1989
Language: English Record Type: Fulltext
Document Type: Newswire; Trade
Word Count: 266

(USE FORMAT 7 FOR FULLTEXT)

TEXT:

Softool Corp, Santa Barbara, California, is now offering its Change & Configuration Control package, CCC, for software development on HP's 9000 Series 300 and 800 systems under HP-UX. The change...

...any language, objects, executable commands and procedures - maintaining an audit trail of changes, the re- creation of prior versions , the management of changes taking place on different versions of the same information, enabling changes to be applied from one evolving version to another and reports on dependencies between components to ensure that all parts of...

? t29/3,k/19

29/3,K/19 (Item 3 from file: 674)
DIALOG(R)File 674:Computer News Fulltext
(c) 2006 IDG Communications. All rts. reserv.

059558
Code Conspirators
Computerworld Intranets
For programmers, collaboration has always been a necessary evil, and cultural biases persist. But leading-edge organizations are finding that shared design elements, knowledge bases and extended development teams made possible by the intranet really work.
Byline: Natalie Engler
Journal: Computerworld Page Number: 1
Publication Date: May 26, 1997
Word Count: 3298 Line Count: 312

Text:
... creates links to the Internet or downloads files that describe standards, best practices, software classes, version control and more. Angelo DiSanto created and maintains the database knowledge center, which contains information about all the databases the company...

... with an E-mail notification system that automatically sends users a message every time a document that concerns them changes . They use a second intranet to deploy executable files - the code that runs the "black ...

... Microsoft Corp.'s Windows NT 4.0 connected to a Microsoft SQL Server database. They developed the interface using HTTP Version 3.0, to provide bidirectional file transfer functionality. The payoff: All the constituent organizations receive...

... They still wrestle with incompatibility issues, such as reconciling the different Unix and Windows file formats and tools. Nor does the intranet help them balance the need for security with the...

... moving its "entire" software development process (source code management, case models, simulation models, requirements documentation, program management, job schedules and change management) to a single database and communicating all that information through a Web server. But ...

... It does all of its "intelligent" programming on a self-contained NT workstation. When an application update is ready, staffers copy it to production and post it to a home page. People...charts and an applet that charts automatically, so they can create static charts that are updated daily. The developers compile data from various sources and publish it as reusable tablets and charts in HTML. Examples include...

... commercially available tools: Version control: Maintains a history of changes to a component as it evolves over time and allows access to a particular version , not just the last version created . Workspace management: Manages independent workspaces for developers so each can work in isolation of other...

...each component on any others, including, where possible, dependencies on nonsoftware items such as design documents or test specs. Change management: Addresses problem tracking and change control and the presentation and analysis of management information...

... log change requests and problem reports and monitor their progress

through the system to closure? Build and release support: Reduces build times dramatically by reusing partially built items from previous builds. Release support allows developers to track which users have which versions of which components and to therefore be sure...

... life cycle phases before being released. Helps ensure that testing and quality assurance occur before release . Application Building Sapient Corp., a 6-year-old, $45 million, Cambridge, Mass.-based consulting firm that designs... worth it." Create a document repository and supplement it with automatic E-mail notification whenever documents change . Be proactive. Get involved with users who are creating Web sites. Educate them on testing...
?

File 348:EUROPEAN PATENTS 1978-2007/ 200708
(c) 2007 European Patent Office
File 349:PCT FULLTEXT 1979-2007/UB=20070301UT=20070222
(c) 2007 WIPO/Thomson

Set	Items	Description
S1	688637	MESSAGE? ? OR DOCUMENT? ? OR FILE OR FILES OR DATAFILE? OR SCHEMA? ? OR SCHEMATA? ? OR DATASTRUCTURE?
S2	139	TEXTMESSAGE? OR TEXTFILE? OR TEXTDATA OR COMPUTERFILE?
S3	1239285	OBJECT? ? OR CONTENT? ?
S4	2956524	APPLICATION? ? OR PROGRAM? ? OR PROGRAMME OR PROGRAMMES OR APP OR APPS OR SOFTWARE? OR SOFT()WARE? ?
S5	120	SOURCECOD? OR PROGRAMCOD? OR PROGRAMMECOD?
S6	88281	PACKET? ? OR DATAPACKET? OR DATAGRAM? OR DATA()GRAM? ?
S7	2306172	UPDAT???? OR UPGRAD? OR UP()(GRAD???? OR DAT????) OR REVIS-???? OR CHANG??? OR MODIFY? OR MODIFIE? ? OR MODIFICAT?
S8	952951	ADAPT??? OR ADAPTATION? OR RECONFIGUR? OR RE()CONFIGUR????-?? OR ALTERR? OR ALTERATION? OR ALTER?? OR ALTERING
S9	32636	REFORMAT? OR REFRESH??? OR RE()(FORMAT???? OR FRESH???)
S10	744860	S7:S9(5N)(S1:S6 OR DATA OR DATUM? ?)
S11	698188	CHAIN???? OR INTERMEDIATE?
S12	168070	MAP OR MAPS OR MAPPED OR MAPPING? ? OR MAPED OR MAPING? ?
S13	158766	ITERAT???? OR PERMUTAT? OR EVOLV??? OR EVOLUTION?
S14	636015	VERSION??? OR ADAPTATION? ? OR REVISION? ? OR EDITION? ? OR RELEASE OR RELEASES
S15	44676	S14(5N)(GENERAT???? OR BUILD??? OR BUILT OR CONSTRUCT???? - OR CREAT???? OR PRODUCE? ? OR PRODUCING OR PRODUCTION? ?)
S16	23877	S14(5N)(PROD? ? OR DERIV??? OR DERIVATION? ? OR PROPAGAT??-?? OR DEFIN??? OR DEFINITION? OR ESTABLISH? OR DEVELOP?????)
S17	43690	S14(5N)(FORM OR FORMS OR FORMED OR FORMING OR FORMATION? ?)
S18	3177	S15:S17(25N)(S11 OR S13)
S19	111	S18(25N)S10
S20	58	S19 AND AC=US/PR AND AY=(1963:1999)/PR
S21	58	S19 AND AC=US AND AY=1963:1999
S22	58	S19 AND AC=US AND AY=(1963:1999)/PR
S23	25	S19 AND PY=1963:1999
S24	66	S20:S23
S25	132	S18(50N)S10
S26	21	S25 NOT S19
S27	10	S26 AND AC=US/PR AND AY=(1963:1999)/PR
S28	10	S26 AND AC=US AND AY=1963:1999
S29	10	S26 AND AC=US AND AY=(1963:1999)/PR
S30	10	S26 AND PY=1963:1999
S31	14	S27:S30

31/5,K/8 (Item 8 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS
(c) 2007 European Patent Office. All rts. reserv.

00274836
Method for file management on a read-only information carrier.
Verfahren zur Dateiverwaltung mit einem Nur-Lese-Informationstrager.
Procede de gestion de fichiers sur un support d'information non-effacable.
PATENT ASSIGNEE:
Picard, Michel, (812540), 16, Cours du Buisson, F-77420 Noisiel, (FR),
(applicant designated states: DE;GB)
INVENTOR:
Picard, Michel, 16, Cours du Buisson, F-77420 Noisiel, (FR)
LEGAL REPRESENTATIVE:
Mongredien, Andre et al (17411), c/o BREVATOME 25, rue de Ponthieu,
F-75008 Paris, (FR)
PATENT (CC, No, Kind, Date): EP 263014 A1 880406 (Basic)
EP 263014 B1 911127
APPLICATION (CC, No, Date): EP 87402075 870917;
PRIORITY (CC, No, Date): FR 8613144 860919
DESIGNATED STATES: DE; GB
INTERNATIONAL PATENT CLASS (V7): G06F-015/40; G11B-020/12;
CITED REFERENCES (EP A):
PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON DATA ENGINEERING, Los Angeles, 24-27 avril 1984, pages 175-180, IEEE, New York, US; P. RATHMANN: "Dynamic data structures on optical disks"
ACM PODS, 1982, pages 239-246, ACM; D. MAIER: "Using write-once memory for database storage"
IBM TECHNICAL DISCLOSURE BULLETIN, vol. 24, no. 1B, juin 1981, pages 775,776, New York, US; D.D. GROSSMAN: "Updating non-erasable storage"
ANGEWANDTE INFORMATIK, vol. 15, no. 1, janvier 1973, pages 21-24, Braunschweig, DE; H.-J. SKRONN: "Verwaltung einer Datenbank durch eine Hierarchie von W|rterb}chern";

ABSTRACT EP 263014 A1
Procede de gestion de fichiers et de documents numerises sur un support d'information non-effacable.
Le procede consiste a memoriser des objets de type en-tete (ET), fichiers (F) et repertoires (R) et une table (TAP, ETAP) pour convertir une adresse logique en une adresse physique.
Selon l'invention, une partition en plusieurs segments est definie sur l'espace de memorisation, chaque segment etant destine a memoriser un seul type d'objet.
La structure choisie pour les objets permet de memoriser les informations suivant une architecture arborescente dont les repertoires forment les noeuds, et les fichiers les noeuds terminaux. La partition est modifiable et est memorisee dans l'en-tete.
Chaque type d'objet comprend des moyens de type champ de chainage permettant une mise a jour de son contenu.
ABSTRACT WORD COUNT: 124

LEGAL STATUS (Type, Pub Date, Kind, Text):
Application: 880406 A1 Published application (A1with Search Report ;A2without Search Report)
Examination: 881102 A1 Date of filing of request for examination: 880910
Examination: 910320 A1 Date of despatch of first examination report: 910204
Grant: 911127 B1 Granted patent
Oppn None: 921119 B1 No opposition filed
LANGUAGE (Publication,Procedural,Application): French; French; French
FULLTEXT AVAILABILITY:
Available Text Language Update Word Count
CLAIMS B (English) EPBBF1 744

CLAIMS B	(German)	EPBBF1	726
CLAIMS B	(French)	EPBBF1	785
SPEC B	(French)	EPBBF1	8045
Total word count - document A			0
Total word count - document B			10300
Total word count - documents A + B			10300

...CLAIMS n))) when the segment (SO) containing successive header versions is full, a new segment is created , said new header version is entered in said new segment and chaining is brought about between said new header version and said present header version.

3. Process...

...in that the objects of the directory type have a chaining field (PTR) permitting an updating of the content of said objects .

4. Process according to any one of the claims 1 to 3 for the management ...

? t24/5,k/5,14,53,54,63

24/5,K/5 (Item 5 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS
(c) 2007 European Patent Office. All rts. reserv.

01116702
INCREMENTAL UPDATING
INKREMENTALES AKTUALISIEREN
MISE A JOUR INCREMENTIELLE
PATENT ASSIGNEE:
SYMANTEC CORPORATION, (1606222), 20330 Stevens Creek Boulevard, Cupertino, CA 95014, (US), (Proprietor designated states: all)
INVENTOR:
NACHENBERG, Carey, S., 19585 Shadow Glen Circle, Northridge, CA 91326, (US)
SOBEL, William, E., 26830 West Wyatt Lane, Stevenson Ranch, CA 91381, (US)
LEGAL REPRESENTATIVE:
Beresford, Keith Denis Lewis et al (28274), 2-5 Warwick Court High Holborn, London WC1R 5DH, (GB)
PATENT (CC, No, Kind, Date): EP 1082651 A2 010314 (Basic)
EP 1082651 B1 030129
WO 99063432 991209
APPLICATION (CC, No, Date): EP 99927158 990602; WO 99US12228 990602
PRIORITY (CC, No, Date): US 89930 980603
DESIGNATED STATES: DE; FR; GB
INTERNATIONAL PATENT CLASS (V7): G06F-009/44
CITED PATENTS (EP B): EP 717353 A
CITED PATENTS (WO A): XP 51138
CITED REFERENCES (EP B):
"USER-FRIENDLY INTERFACE FOR APPLYING SPECIFIC LEVELS OF COORDINATED UPDATES TO SOFTWARE IN THE FIELD" IBM TECHNICAL DISCLOSURE BULLETIN,US,IBM CORP. NEW YORK, vol. 31, no. 10, page 258-259 XP000051138 ISSN: 0018-8689;
CITED REFERENCES (WO A):
"USER-FRIENDLY INTERFACE FOR APPLYING SPECIFIC LEVELS OF COORDINATED UPDATES TO SOFTWARE IN THE FIELD" IBM TECHNICAL DISCLOSURE BULLETIN,US,IBM CORP. NEW YORK, vol. 31, no. 10, page 258-259 XP000051138 ISSN: 0018-8689;
NOTE:
No A-document published by EPO
LEGAL STATUS (Type, Pub Date, Kind, Text):

Application:	010314	A2 Published application without search report
Application:	20000202	A2 International application. (Art. 158(1))
Oppn None:	040121	B1 No opposition filed: 20031030
Change:	030115	A2 Inventor information changed: 20021127
Change:	010516	A2 Inventor information changed: 20010328
Examination:	010314	A2 Date of request for examination: 20001221
Examination:	010509	A2 Date of dispatch of the first examination report: 20010326
Change:	020724	A2 Title of invention (English) changed: 20020605
Change:	020724	A2 Title of invention (French) changed: 20020605
Grant:	030129	B1 Granted patent
Application:	20000202	A2 International application entering European phase

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS B	(English)	200305	1654
CLAIMS B	(German)	200305	1563
CLAIMS B	(French)	200305	1793
SPEC B	(English)	200305	3038

Total word count - document A 0

Total word count - document B 8048
Total word count - documents A + B 8048

...SPECIFICATION the current version. A previously published hub version can be used with the two incremental updates (110, 112) to produce a file of the current version. The first incremental update (110) is used (316) to produce the most recent hub version , and the second incremental update (112) is used (322) to produce the current version .

When such an update is carried out, an intermediate version of the file, the most recent hub version , is produced (316). The file of the most recent hub version is retained (324) locally subsequent to...

24/5,K/14 (Item 14 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS
(c) 2007 European Patent Office. All rts. reserv.

00546088
Improved multiple version database concurrency control system.
Gleichzeitigkeitssteuerungssystem fur eine Datenbank mit mehreren Versionen.
Systeme de commande de simultaneite pour une base de donnees avec des versions multiples.
PATENT ASSIGNEE:
International Business Machines Corporation, (200120), Old Orchard Road, Armonk, N.Y. 10504, (US), (applicant designated states: DE;FR;GB)
INVENTOR:
Lorie, Raymond Amand, 1272 Echo Valley Drive, San Jose, California 95120, (US)
Mohan, Chandrasekaran, 727 Portswood Drive, San Jose, California 95120, (US)
Pirahesh, Mir Hamid, 1282 Quail Creek Circle, San Jose, California 95120, (US)
LEGAL REPRESENTATIVE:
Burt, Roger James, Dr. (52152), IBM United Kingdom Limited Intellectual Property Department Hursley Park, Winchester Hampshire SO21 2JN, (GB)
PATENT (CC, No, Kind, Date): EP 544517 A2 930602 (Basic)
EP 544517 A3 931027
APPLICATION (CC, No, Date): EP 92310783 921125;
PRIORITY (CC, No, Date): US 801769 911126
DESIGNATED STATES: DE; FR; GB
INTERNATIONAL PATENT CLASS (V7): G06F-015/403;
CITED PATENTS (EP A): WO 8400426 A; EP 501180 A
CITED REFERENCES (EP A):
IEEE PROCEEDINGS OF THE 10TH INTERNATIONAL PHOENIX CONFERENCE ON COMPUTERS AND COMMUNICATIONS, CH2859-5/91, 30 March 1991, SCOTTSDALE, ARIZONA pages 130 - 136 KATAOKA R. ET AL. 'A Multiversion Concurrency Control Algorithm for Concurrent Execution of Partial Update and Bulk Retrieval Transactions';

ABSTRACT EP 544517 A2
An improved concurrency control system for application to a distributed concurrent transaction and query processing system using multi-version database records to overcome delays arising from lock conflicts. Read-only queries are afforded a consistent "stable state" of the database during the life of the query. Updating transactions requiring locks can proceed without waiting for the termination of long queries. At least two database versions are necessary, although availability of more versions permits long read-only queries to phase-out over time without forcing new queries to use aged "stable-state" data and without roll-back. Read-only queries can be terminated and converted to locking transactions to permit an update of the "stable state" database version before the queries would normally terminate. A novel record key structure

having a plurality of substructures corresponding to the several database versions is used to access database records. Rapid selection of proper record version and efficient version tracking and updating is effected using several bit-mapped transaction index tables. (see image in original document)
ABSTRACT WORD COUNT: 165

LEGAL STATUS (Type, Pub Date, Kind, Text):
 Application: 930602 A2 Published application (A1with Search Report ;A2without Search Report)
 Change: 931006 A2 Representative (change)
 Search Report: 931027 A3 Separate publication of the European or International search report
 Examination: 931110 A2 Date of filing of request for examination: 930918
 Withdrawal: 961120 A2 Date on which the European patent application was withdrawn: 960927
LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:
Available Text Language Update Word Count
 CLAIMS A (English) EPABF1 685
 SPEC A (English) EPABF1 9282
Total word count - document A 9967
Total word count - document B 0
Total word count - documents A + B 9967

...SPECIFICATION by a read-only query remains unchanged during the life of the query.
A new version of a record is created whenever any updating transaction writes new data to a record that was created by a previous committed transaction. An arbitrarily long chain of increasingly outdated "stable-state" versions can be created and maintained. Each "stable-state" version need be retained only as long as necessary to terminate all active queries that are...

24/5,K/53 (Item 34 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT
(c) 2007 WIPO/Thomson. All rts. reserv.

00757044 **Image available**
PROCESS CONTROL CONFIGURATION SYSTEM WITH PARAMETERIZED OBJECTS
SYSTEME DE CONFIGURATION DE COMMANDE DE PROCESSUS VIA DES OBJETS PARAMETRES
Patent Applicant/Assignee:
THE FOXBORO COMPANY, 33 Commercial Street, Foxboro, MA 02035, US, US (Residence), US (Nationality)
Inventor(s):
DARDINSKI Steven, 7 Vose Hill Road, Westford, MA 01886, US
CAMINO Nestor, 4 Blue Sky Drive, Hingham, MA 02043, US
ELDRIDGE Keith, 239 Poquanticut Avenue, North Easton, MA 02356, US
HALL Robert, 37 Dean Street, South Easton, MA 02375, US
JOHNSON Mark, 254 Old Wood Road South, North Attleboro, MA 02760, US
MACKAY Brian, 335 Cove Drive, Coppell, TX 75019-5679, US
MESKONIS Paul, 178 Rock Street, Norwood, MA 02062, US
SHERRILL Tom, 220 Landry Avenue, North Attleboro, MA 02760, US
VOLK Scott, 25 Ramblewood Drive, North Easton, MA 02356, US
Legal Representative:
POWSNER David J, Nutter, McClennen & Fish, LLP, One International Place, Boston, MA 02110-2699, US
Patent and Priority Information (Country, Number, Date):
Patent: WO 200070417 A1 20001123 (WO 0070417)
Application: WO 2000US13618 20000517 (PCT/WO US0013618)
Priority Application: US 99134597 19990517; US 99448374 19991123; US 99448845 19991123; US 99448223 19991123

Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AE AL AM AT AU AZ BA BB BG BR BY CA CH CN CR CU CZ DE DK DM EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG UZ VN YU ZA ZW
(EP) AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE
(OA) BF BJ CF CG CI CM GA GN GW ML MR NE SN TD TG
(AP) GH GM KE LS MW SD SL SZ TZ UG ZW
(EA) AM AZ BY KG KZ MD RU TJ TM
Main International Patent Class (v7): G05B-015/00
International Patent Class (v7): G05B-019/18
Publication Language: English
Filing Language: English
Fulltext Availability:
Detailed Description
Claims
Fulltext Word Count: 94824

English Abstract
A workstation (11) that is coupled to one or more controllers (10A & 10B) on which reside process control systems for monitoring and/or controlling one or more processes (12). Server (16) represents an optional additional source of classes defining objects for modeling a control system and for configuring controllers (10A & 10B). Network (14) provides a communications medium permitting the downloading of control algorithms and other configuration information to controllers (10A & 10B).

French Abstract
L'invention concerne un poste de travail (11) couple a une ou plusieurs unites de commande (10A, 10B) accueillant des systemes de commande de processus qui permettent de surveiller et/ou de commander un ou plusieurs processus (12). Un serveur (16) represente une source additionnelle facultative de classes definissant des objets pour la modelisation d'un systeme de commande et pour la configuration des unites de commande (10A, 10B). Un reseau (14) tient lieu de support de communication permettant le telechargement d'algorithmes de commande et autres informations de configuration vers les unites de commande (10A, 10B).

Legal Status (Type, Date, Text)
Publication 20001123 A1 With international search report.
Publication 20001123 A1 Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.
Examination 20010315 Request for preliminary examination prior to end of 19th month from priority date

Fulltext Availability:
Detailed Description

Detailed Description
... checked-in version of a Parameterized Object has a one-to-one association with a Change Description object , which records what modification (s) were made to the object in order to create the associated revision .

A checked-in version of a Parameterized Object maintains a reference to its Previous Version, in support of the requirement to be able to " chain " down the version tree of any Parameterized Object. This would allow an application to get...

24/5,K/54 (Item 35 from file: 349)

DIALOG(R)File 349:PCT FULLTEXT
(c) 2007 WIPO/Thomson. All rts. reserv.

00738348

ELECTRONIC COMMERCE SYSTEMS AND PROCESSES, ESPECIALLY FOR THE CABLE TELEVISION INDUSTRY

SYSTEMES ET PROCEDES DE COMMERCE ELECTRONIQUE, SPECIALEMENT DESTINES A L'INDUSTRIE DE LA TELEVISION PAR CABLE

Patent Applicant/Assignee:
VIDEO NETWORKS INCORPORATED, 245 Hembree Park Drive, Suite 100, Roswell, GA 30076, US, US (Residence), US (Nationality), (For all designated states except: US)
Patent Applicant/Inventor:
FABIANO Leonard J III, 3525 River Ferry Drive, Alpharetta, GA 30022, US, US (Residence), US (Nationality), (Designated only for: US)
Legal Representative:
PRATT John S, Kilpatrick Stockton LLP, 1100 Peachtree Street, Suite 2800, Atlanta, GA 30309-4530, US
Patent and Priority Information (Country, Number, Date):
Patent: WO 200051335 A2 20000831 (WO 0051335)
Application: WO 2000US4817 20000225 (PCT/WO US0004817)
Priority Application: US 99122136 19990226
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
CA MX US
Main International Patent Class (v7): H04N
Publication Language: English
Filing Language: English
Fulltext Availability:
Detailed Description
Claims
Fulltext Word Count: 27266

English Abstract
Electronic commerce systems and processes which are adapted to manage workflow, information and product distribution in transactions among various parties in a supply chain. The systems and processes may rely on a common document model which allows each of the entities to maintain and use its own legacy or proprietary systems and interfaces, but yet to communicate and transact with each other through a master platform or system which handles data with associated metadata, such as via a common document model, so that all platforms have ready, efficient and easily translatable or transformable access to common information. Such systems and processes are particularly well suited for purchasing, distributing, payment and other transactions in the cable television industry, including transactions associated with advertising and commercials.

French Abstract
L'invention concerne des systemes et des procedes de commerce electronique, qui sont adaptes pour gerer le deroulement des operations, les informations et la distribution des produits au cours de transactions entre differentes parties dans une chaine de distribution. Les systemes et procedes peuvent dependre d'un modele de document commun qui permet a chaque entite de mettre a jour et d'utiliser ses propres interfaces et systemes existants ou exclusifs, mais qui exige de communiquer et de faire des transactions entre les parties par une plate-forme maitresse ou un systeme qui traite des donnees et des metadonnees associees, comme via un modele de document commun, de facon que toutes les plate-formes soient dotees d'un acces aux informations communes rapide, efficace et facilement traduit ou transforme. Ces systemes et procedes sont particulierement bien adaptes pour l'acquisition, la distribution, le paiement et d'autres transactions dans l'industrie de la television par cable, y compris des transactions associees a la publicite et aux

messages publicitaires.

Legal Status (Type, Date, Text)
Publication 20000831 A2 Without international search report and to be republished upon receipt of that report.
Search Rpt 20010111 Late publication of international search report

Fulltext Availability:
Detailed Description

Detailed Description
... and stored in Document Definition Files (DDF). Each will dictate the structure of a single version of a form document. This structure allows document structures to evolve and still provide usability of older versions of documents. Each time the form of a document is altered , a new DDF must be produced so that the system utilities can decipher its contents...

24/5,K/63 (Item 44 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT
(c) 2007 WIPO/Thomson. All rts. reserv.

00358799 **Image available**
METHOD FOR COLLECTING, MANAGING, MANIPULATING, AND CHECKING DATA DURING CONSTRUCTION OF A VIRTUAL ENVIRONMENT
PROCEDE DE COLLECTE, GESTION, MANIPULATION ET VERIFICATION DE DONNEES AU COURS DE LA CONSTRUCTION D'UN ENVIRONNEMENT VIRTUEL
Patent Applicant/Assignee:
VIRTUAL WORLD ENTERTAINMENT INC,
Inventor(s):
McCOY David S,
ALBERTSON John-Mark,
Patent and Priority Information (Country, Number, Date):
Patent: WO 9641313 A1 19961219
Application: WO 96US10032 19960606 (PCT/WO US9610032)
Priority Application: US 95480649 19950607
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AU CA JP MX AT BE CH DE DK ES FI FR GB GR IE IT LU MC NL PT SE
Main International Patent Class (v7): G06T-011/40
Publication Language: English
Fulltext Availability:
Detailed Description
Claims
Fulltext Word Count: 11489

English Abstract
A method for collecting, managing, manupulating, and checking data, during construction of a virtual environment, and automatically re-processing the subset of data necessary to produce a resource for use by a simulation program, is provided. This method provides for the repeated application of a designated material to commonly designated elements of multiple objects (114). Once the material has been designated (110) to be applied to a particular element of an object (110), application to other objects is facilitated by designating the common element (102, 104). This method significantly reduces the amount of repetitive artist labor related to iterated operations performed and features improved error detection and indication.

French Abstract
L'invention se rapporte a un procede de collecte, gestion, manipulation et verification de donnees au cours de la construction d'un environnement

virtuel, et de retraitement automatique du sous-fichier de donnees necessaire pour une generation de ressource destinee a etre utilisee par un programme de simulation. Ce procede prevoit l'application repetee d'un materiau designe a des elements d'objets multiples (114). Apres la designation du materiau (110) a appliquer a un element particulier d'un objet (110), on facilite l'application a d'autres objets en designant l'element commun (102, 104). Ce procede reduit considerablement la quantite de travail repetitif d'un dessinateur concernant des operations repetitives et permet une meilleure detection et une meilleure signalisation d'erreurs.

Patent and Priority Information (Country, Number, Date):
Patent: ... 19961219
Fulltext Availability:
Detailed Description
Publication Year: 1996

Detailed Description
... to the number of
revisions it undergoes during its transition from source data file to intermediate data file to finished data file, it is of paramount importance that the duration of this
construction and revision process be minimized to permit sufficient time for error checking and revisions
OBJECTS AND SUMMARY OF THE INVENTION
In the present invention, a means for collecting, managing, manipulating...
?

09/295,690

```
File   2:INSPEC 1898-2007/Feb W4
         (c) 2007 Institution of Electrical Engineers
File   6:NTIS 1964-2007/Mar W1
         (c) 2007 NTIS, Intl Cpyrght All Rights Res
File   8:Ei Compendex(R) 1884-2007/Feb W4
         (c) 2007 Elsevier Eng.  Info. Inc.
File  34:SciSearch(R) Cited Ref Sci 1990-2007/Feb W4
         (c) 2007 The Thomson Corp
File  35:Dissertation Abs Online 1861-2007/Feb
         (c) 2007 ProQuest Info&Learning
File  65:Inside Conferences 1993-2007/Mar 07
         (c) 2007 BLDSC all rts. reserv.
File  94:JICST-EPlus 1985-2007/Mar W2
         (c)2007 Japan Science and Tech Corp(JST)
File  95:TEME-Technology & Management 1989-2007/Mar W1
         (c) 2007 FIZ TECHNIK
File  99:Wilson Appl. Sci & Tech Abs 1983-2007/Feb
         (c) 2007 The HW Wilson Co.
File 144:Pascal 1973-2007/Feb W4
         (c) 2007 INIST/CNRS
File 256:TecInfoSource 82-2007/Oct
         (c) 2007 Info.Sources Inc
File 266:FEDRIP 2007/Jan
         Comp & dist by NTIS, Intl Copyright All Rights Res
File 434:SciSearch(R) Cited Ref Sci 1974-1989/Dec
         (c) 2006 The Thomson Corp
File 583:Gale Group Globalbase(TM) 1986-2002/Dec 13
         (c) 2002 The Gale Group
File  56:Computer and Information Systems Abstracts 1966-2007/Feb
         (c) 2007 CSA.
File  60:ANTE: Abstracts in New Tech & Engineer 1966-2007/Feb
         (c) 2007 CSA.

Set     Items   Description
S1    1089649   MESSAGE? ? OR DOCUMENT? ? OR FILE OR FILES OR DATAFILE? OR
            SCHEMA? ? OR SCHEMATA? ? OR DATASTRUCTURE?
S2         49   TEXTMESSAGE? OR TEXTFILE? OR TEXTDATA OR COMPUTERFILE?
S3    2953161   OBJECT? ? OR CONTENT? ?
S4    9630220   APPLICATION? ? OR PROGRAM? ? OR PROGRAMME OR PROGRAMMES OR
              APP OR APPS OR SOFTWARE? OR SOFT()WARE? ?
S5        127   SOURCECOD? OR PROGRAMCOD? OR PROGRAMMECOD?
S6     216475   PACKET? ? OR DATAPACKET? OR DATAGRAM? OR DATA()GRAM? ?
S7    8203828   UPDAT???? OR UPGRAD? OR UP()(GRAD???? OR DAT????) OR REVIS-
            ???? OR CHANG??? OR MODIFY? OR MODIFIE? ? OR MODIFICAT?
S8    2369119   ADAPT??? OR ADAPTATION? OR RECONFIGUR? OR RE()CONFIGUR????-
            ?? OR ALTERR? OR ALTERATION? OR ALTER?? OR ALTERING
S9      31591   REFORMAT? OR REFRESH??? OR RE()(FORMAT???? OR FRESH???)
S10    523326   S7:S9(5N)(S1:S6 OR DATA OR DATUM? ?)
S11   2359564   CHAIN???? OR INTERMEDIATE?
S12   1253462   MAP OR MAPS OR MAPPED OR MAPPING? ? OR MAPED OR MAPING? ?
S13   2210340   ITERAT???? OR PERMUTAT? OR EVOLV??? OR EVOLUTION?
S14   1949413   VERSION??? OR ADAPTATION? ? OR REVISION? ? OR EDITION? ? OR
             RELEASE OR RELEASES
S15     61826   S14(5N)(GENERAT???? OR BUILD??? OR BUILT OR CONSTRUCT???? -
            OR CREAT???? OR PRODUCE? ? OR PRODUCING OR PRODUCTION? ?)
S16     71874   S14(5N)(PROD? ? OR DERIV??? OR DERIVATION? ? OR PROPAGAT??-
            ?? OR DEFIN??? OR DEFINITION? OR ESTABLISH? OR DEVELOP?????)
S17     38631   S14(5N)(FORM OR FORMS OR FORMED OR FORMING OR FORMATION? ?)
S18     17678   S15:S17 AND S11:S13
S19      1078   S18 AND S10
S20     14220   S15:S17 AND (S11 OR S13)
S21       777   S20 AND S10
S22      9977   S15:S17(S)(S11 OR S13)
S23       607   S22 AND S10
```

S24	354916	S7:S9(5N)S1:S6
S25	8362	S15:S16(S)(S11 OR S13)
S26	465	S25 AND S24
S27	274	S26/1999:2007
S28	191	S26 NOT S27
S29	135	RD (unique items)
S30	45	S23 AND (FORMAT?? OR FORMATT? OR FORMATING)
S31	23	S30/2000:2007
S32	22	S30 NOT S31
S33	13	RD (unique items)
S34	125	S29 NOT S33

? t34/7/1,7,10,14-15,18,20,24,26,28,38,52,81,90,101,110

34/7/1 (Item 1 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

07188838 INSPEC Abstract Number: C1999-04-7420-032
Title: Lessons from experimenting model-based requirements engineering for software-intensive railway control systems
Author(s): Callet, S.; Moinon, C.; Ermine, F.; Rouge, A.
Author Affiliation: Matra Transp. Int., Montrouge, France
Conference Title: CONQUEST '98. Conference on Quality Engineering in Software Technology p.27-36
Publisher: ASQF, Erlangen, Germany
Publication Date: 1998 Country of Publication: Germany 221 pp.
Material Identity Number: XX-1998-02466
Conference Title: Proceedings of CONQUEST-2. Second Conference on Quality Engineering in Software Technology
Conference Date: 28-29 Sept. 1998 Conference Location: Nurnberg, Germany
Availability: ASQF-Programmkomitee, Wetterkreuz 19a, 91058 Erlangen, Germany
Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: In the software industry, it has been recognized that more of 50% of errors can be requirements errors and that solving them early can be up to several hundreds of times cheaper than solving them late. Moving to a more anticipating mode however, requires adoption of a systems engineering process on top of the software development process, as well as a systematic requirements engineering approach. The paper presents the experimentation of a particular instance of such an approach, known as model based requirements analysis at the systems level of a railway control and command project. More precisely, it describes the intermediate lessons drawn from the FRUITS experiment that aims at assessing the value of model based requirements engineering against traditional documentation driven requirement engineering. The experiment is carried out in parallel with an operational project that produced a sequence of different versions of the requirements document. The experiment took one of the versions as a starting point (the initial version) and revised it by applying model based requirements analysis techniques to produce the next version (the revised version). After positioning the document based and model based approaches to requirements analysis, we start by explaining the main concepts underlying the chosen model based approach. Then we make an assessment of the revised version against the initial version of the requirements document and we also report some of the lessons learned by the systems engineers involved in the experiment. (4 Refs)
Subfile: C
Copyright 1999, IEE

34/7/7 (Item 7 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

06719860 INSPEC Abstract Number: C9711-6160D-015
Title: Schema versioning for multitemporal relational databases
Author(s): De Castro, C.; Grandi, F.; Scalas, M.R.
Author Affiliation: CNR, Bologna, Italy
Journal: Information Systems vol.22, no.5 p.249-90
Publisher: Elsevier,
Publication Date: July 1997 Country of Publication: UK
CODEN: INSYD6 ISSN: 0306-4379
SICI: 0306-4379(199707)22:5L.249:SVMR;1-O
Material Identity Number: I275-97005

U.S. Copyright Clearance Center Code: 0306-4379/97/$17.00+0.00
Document Number: S0306-4379(97)00017-3
Language: English Document Type: Journal Paper (JP)
Treatment: Practical (P); Theoretical (T)
Abstract: In order to follow the evolution of application needs, a database management system is easily expected to undergo changes involving database structure after implementation. Schema evolution concerns the ability of maintaining extant data in response to changes in database structure. Schema versioning enables the use of extensional data through multiple schema interfaces as created by a history of schema changes . We consider schema versioning in a broader perspective and introduce new design options whose distinct semantic properties and functionalities are discussed. First of all, we consider solutions for schema versioning along transaction time but also along valid time. Moreover, the support of schema versioning implies operations both at intensional and extensional level. Two distinct design solutions (namely single- and multi-pool) are presented for the management of extensional data in a system supporting schema versioning. Finally, a further distinction is introduced to define synchronous and asynchronous management of versioned data and schemata. The proposed solutions differ in their semantics and in the possible operations they support. The mechanisms for the selection of data through a schema version are in many cases strictly related to the particular schema versioning solution adopted, that also affects the data definition and manipulation language at user interface level. In particular, we show how the temporal language TSQL2, originally designed to support basic functionalities of transaction-time schema versioning, can accordingly be extended. (23 Refs)
Subfile: C
Copyright 1997, IEE

34/7/10 (Item 10 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

06465848 INSPEC Abstract Number: C9702-6160J-010
Title: Maintaining database consistency in the presence of schema evolution
Author(s): Ling Liu
Author Affiliation: Dept. of Comput. Sci., Alberta Univ., Edmonton, Alta., Canada
Conference Title: Database Applications Semantics. Proceedings of the IFIP WG 2.6 Working Conference on Database Applications Semantics (DS-6) p.549-72
Editor(s): Meersman, R.; Mark, L.
Publisher: Chapman & Hall, London, UK
Publication Date: 1997 Country of Publication: UK vii+610 pp.
ISBN: 0 412 72600 9 Material Identity Number: XX96-03673
Conference Title: Database Applications Semantics. Proceeding of the IFIP WG 2.6 Working Conference on Database Applications Semantics (DS-6)
Conference Sponsor: IFIP
Conference Date: 30 May-2 June 1995 Conference Location: Atlanta, GA, USA
Language: English Document Type: Conference Paper (PA)
Treatment: Theoretical (T)
Abstract: With rapid advances in computer network technology and the increasing interest in global information sharing, grows the need for facilities that can effectively maintain the database consistency and program compatibility as the scheme evolves . The author presents a general framework based on versions of schema for supporting seamless schema evolution in large-scale object-oriented software systems. They argue that the effectiveness of using the schema version model to materialize schema evolution relies not only on the management of version derivation of schema, but also on the ability to maintain

consistency of the database instances with the new schema versions, and the flexibility of sharing instances among versions of schema. Semantics of scheme versioning is studied with the objective to facilitate instance adaptation and program compatibility in the presence of schema updates . A number of options for sharing of instances among versions of a scheme is developed . Applications may derive versions of schema upon requests for schema updates , and define the instance access scope (the set of objects that are accessible under a given schema version) for each new version of scheme by either creating their own instance access scope or inheriting the instance access scope of its ancestor schema versions in terms of a selection of options. The significance of our approach is; the abilities for maintaining database consistency, in the presence of schema modification , without irreversibly changing the objects that exist before the schema modification , and; the facilities that allow multi-users and applications to create and manipulate different collections of objects under different versions of schema. As a consequence, many organizational investments of the existing customer set, such as application design and development, can remain operational in the presence of schema changes . (14 Refs)
Subfile: C
Copyright 1997, IEE

34/7/14 (Item 14 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

06263565 INSPEC Abstract Number: C9606-6160J-008
Title: Database schema evolution using EVER diagrams
Author(s): Chien-Tsai Liu; Shi-Kuo Chang; Chrysanthis, P.K.
Author Affiliation: Dept. of Comput. Sci., Pittsburgh Univ., PA, USA
Conference Title: Proceedings of the Workshop on Advanced Visual Interfaces AVI `94 p.123-32
Editor(s): Catarci, T.; Costabile, M.F.; Levialdi, S.; Sanctucci, G.
Publisher: ACM, New York, NY, USA
Publication Date: 1994 Country of Publication: USA x+249 pp.
ISBN: 0 89791 733 2 Material Identity Number: XX96-01200
U.S. Copyright Clearance Center Code: 0 89791 733 2/94/0010.$3.50
Conference Title: Proceedings of the Workshop on Advanced Visual Interfaces AVI
Conference Date: 1-4 June 1994 Conference Location: Bari, Italy
Language: English Document Type: Conference Paper (PA)
Treatment: Theoretical (T)
Abstract: We present an approach to schema evolution through changes to the ER diagram representing the schema, of a database. In order to facilitate changes to the ER schema we enhance the graphical constructs used in ER diagrams, and develop EVER, an EVolutionary ER diagram for specifying the derivation relationships between schema versions , relationships among attributes, and the conditions for maintaining consistent views of programs. In this paper, we demonstrate the mapping of the EVER diagram into an underlying database and the construction of database views for schema versions . Through the reconstruction of views after database reorganization, changes to an ER diagram can be made transparent to the application programs while all objects in the database remain accessible to the application programs. The EVER system can serve as a front-end for object-oriented databases. (16 Refs)
Subfile: C
Copyright 1996, IEE

34/7/15 (Item 15 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

06250281 INSPEC Abstract Number: C9606-6160J-005
Title: Consistent view removal in transparent schema evolution systems
Author(s): Crestana-Taube, V.; Rundensteiner, E.A.
Author Affiliation: Dept. of Electr. Eng. & Comput. Sci., Michigan Univ., Ann Arbor, MI, USA
Conference Title: Proceedings. Sixth International Workshop on Research Issues in Data Engineering. Interoperability of Nontraditional Database Systems (Cat. No.96TB100021) p.138-47
Publisher: IEEE Comput. Soc. Press, Los Alamitos, CA, USA
Publication Date: 1996 Country of Publication: USA ix+171 pp.
ISBN: 0 8186 7289 7 Material Identity Number: XX96-01105
U.S. Copyright Clearance Center Code: 0 8186 7289 7/96/$05.00
Conference Title: Proceedings RIDE '96. Sixth International Workshop on Research Issues in Data Engineering
Conference Sponsor: IEEE Comput. Soc. Tech. Committee on Data Eng
Conference Date: 26-27 Feb. 1996 Conference Location: New Orleans, LA, USA
Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P); Theoretical (T)
Abstract: We have developed the transparent schema evolution (TSE) system that, simulating schema evolution using object-oriented views, allows for the interoperability of applications with diverse and even changing requirements. TSE relieves users of the risk of making existing application programs obsolete when run against the modified schema, because the old view schema is maintained while a new view schema is generated to capture the changes desired by the user. However TSE may be generating a large number of schema versions (object-oriented view schemata) over time, resulting in an excessive build-up of classes and underlying object instances-some of which may potentially no longer be in use. We propose to solve this problem by developing techniques for effective and consistent schema removal. First, we characterize four potential problems of schema consistency that could be caused by removal of a single virtual class; and then outline our solution approach for each of these problems. Second, we demonstrate that view schema removal is sensitive to the order in which individual classes are processed. Our solution to this problem is the development of a dependency graph model for capturing the class relationships, used as a foundation for selecting among removal sequences. Designed to optimize the performance of the TSE system by effective schema version removal, the proposed techniques will enable more effective interoperability among evolving software applications. (13 Refs)
Subfile: C
Copyright 1996, IEE

34/7/18 (Item 18 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

05986293 INSPEC Abstract Number: C9508-6160-004
Title: Database schema evolution through the specification and maintenance of changes on entities and relationships
Author(s): Chien-Tsai Liu; Chrysanthis, P.K.; Shi-Kuo Chang
Author Affiliation: Dept. of Comput. Sci., Pittsburgh Univ., PA, USA
p.132-51
Editor(s): Loucopoulos, P.
Publisher: Springer-Verlag, Berlin, Germany
Publication Date: 1994 Country of Publication: West Germany xiii+579 pp.
ISBN: 3 540 58786 1
Conference Title: Entity-Relationship Approach - ER '94 Business Modelling and Re-Engineering. 13th International Conference on the Entity-Relationship Approach
Conference Date: 13-16 Dec. 1994 Conference Location: Manchester, UK

Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: A flexible database system needs to support changes to its schema in order to facilitate the requirements of new applications and to support interoperability within a multidatabase system. We present an approach to schema evolution through changes to the entity relationship (ER) schema of a database. We enhance the graphical constructs used in ER diagrams, and develop EVER, an evolutionary ER diagram for specifying the derivation relationships between schema versions , relationships among attributes, and the conditions for maintaining consistent views of programs. Algorithms are presented for mapping the EVER diagram into the underlying database and constructing database views for schema versions . Through the reconstruction of views after database reorganization, changes to an ER diagram can be made transparent to the application programs while all objects in the database remain accessible to the application programs. (24 Refs)
Subfile: C
Copyright 1995, IEE

34/7/20 (Item 20 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

05700784 INSPEC Abstract Number: C9408-6160D-009
Title: A model for schema evolution in temporal relational databases
Author(s): Scalas, M.R.; Cappelli, A.; De Castro, C.
Author Affiliation: Dipartimento di Elettronica, Inf. e Sistemistica, CIOC-CNR, Bologna, Italy
p.223-31
Editor(s): Croisier, A.; Israel, M.; Chavand, F.
Publisher: IEEE Comput. Soc. Press, Los Alamitos, CA, USA
Publication Date: 1993 Country of Publication: USA xiii+541 pp.
ISBN: 0 8186 4030 8
U.S. Copyright Clearance Center Code: 0 8186 4030 8/93/$03.00
Conference Title: Proceedings of COMPEURO '93
Conference Sponsor: IEEE; SEE
Conference Date: 24-27 May 1993 Conference Location: Pris-Evry, France
Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: In temporal databases, transaction time and valid time are added to the data. In this context, the modification of the schema which a database management system (DBMS) can undergo during its lifetime, known as schema changes , must develop into the higher concept of scheme evolution . Changes to the schema produce versions of previous schemas along both time axes and these versions must be maintained and managed during the whole database life. An overview on the issues of temporal relational databases and relational schema changes is presented. A basic outline for the integration of these two aspects into a temporal relational database model supporting schema evolution is given. (7 Refs)
Subfile: C

34/7/24 (Item 24 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

05454458 INSPEC Abstract Number: C9309-6160-015
Title: Management of schema evolution in databases
Author(s): Andany, J.; Leonard, M.; Palisser, C.
Author Affiliation: CUI, Geneve Univ., Switzerland
Conference Title: Proceedings of the Seventeenth International Conference on Very Large Data Bases p.161-70

Editor(s): Lohman, G.M.; Sernadas, A.; Camps, R.
Publisher: Morgan Kaufmann, San Mateo, CA, USA
Publication Date: 1991 Country of Publication: USA xii+596 pp.
Conference Sponsor: IEEE
Conference Date: 3-6 Sept. 1991 Conference Location: Barcelona, Spain
Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: The paper presents a version model which handles database schema changes and which takes evolution into account. Its originality is in allowing the development of partial schema versions , or views of a schema. These versions are created in the same database from a common schema. The authors define the set of authorised modifications on a schema and the rules which guarantee its coherence after transformation. Mechanisms allowing data to be associated with each version are also integrated in the model. (28 Refs)
Subfile: C

34/7/26 (Item 26 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

05286296 INSPEC Abstract Number: C9301-6160J-002
Title: An extensible object-oriented database testbed
Author(s): Morsi, M.M.; Navathe, S.B.; Hyoung-Joo Kim
Author Affiliation: Coll. of Comput., Georgia Inst. of Technol., Atlanta, GA, USA
Conference Title: Eighth International Conference on Data Engineering (Cat. No.92CH3097-3) p.150-7
Publisher: IEEE Comput. Soc. Press, Los Alamitos, CA, USA
Publication Date: 1992 Country of Publication: USA xvi+660 pp.
ISBN: 0 8186 2545 7
U.S. Copyright Clearance Center Code: 0 8186 2545 7/92$3.00
Conference Sponsor: IEEE
Conference Date: 2-3 Feb. 1992 Conference Location: Tempe, AZ, USA
Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: The authors describe the object-oriented design and implementation of an extensible schema manager for object-oriented databases. The open class hierarchy approach has been adopted to achieve the extensibility of the implementation. In this approach. the system meta information is implemented as objects of system classes. A graphical interface for an object-oriented database scheme environment, GOOSE, has been developed. GOOSE supports several advanced features which include schema evolution , schema versioning, and DAG (direct acyclic graph) rearrangement view of a class hierarchy. Schema evolution is the ability to make a variety of changes to a database scheme without reorganization. Schema versioning is the ability to define multiple scheme versions and to keep track of schema changes . A novel type of view for object-oriented databases, the DAG rearrangement view of a class hierarchy, is also supported. (14 Refs)
Subfile: C

34/7/28 (Item 28 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

04923616 INSPEC Abstract Number: C91048603
Title: A persistent store for large shared knowledge bases
Author(s): Mays, E.; Lanka, S.; Dionne, B.; Weida, R.
Author Affiliation: IBM Thomas J. Watson Res. Center, Yorktown Heights, NY, USA
Journal: IEEE Transactions on Knowledge and Data Engineering vol.3,

no.1 p.33-41
Publication Date: March 1991 Country of Publication: USA
ISSN: 1041-4347
U.S. Copyright Clearance Center Code: 1041-4347/91/0300-0033$01.00
Language: English Document Type: Journal Paper (JP)
Treatment: Practical (P)
Abstract: A knowledge-base management system (KBMS) with the following goals is proposed: (1) to allow a knowledge engineer to update a knowledge base and have these updates persist on secondary storage, (2) to allow multiple knowledge engineers to have shared access to a knowledge base and modify the knowledge base concurrently, and (3) to maintain consistency of the shared knowledge base as it evolves . At the heart of the KBMS is the version store, which is a persistent storage structure that maintains multiple versions of a knowledge base. Retrieve and update operations are defined on the version store to efficiently access and modify any version. Objects in the version store are clustered to support efficient access of an entire version or some subparts of it. To store the effects of an update requires space proportional to the size of the update and which is independent of the size of the knowledge base. The additional cost of maintaining multiple versions is within a small constant factor of maintaining a single version. (14 Refs)
Subfile: C

34/7/38 (Item 38 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

03187650 INSPEC Abstract Number: C84009239
Title: GALAAD: a software management system
Author(s): Campbell, I.; Stilling, C.
Conference Title: Productivite et Informatique: Pour une Entreprise Dynamique. Recueil des Conferences du Printemps Convention (Productivity and Data Processing: Two Essentials for a Dynamic Company. Proceedings of the Spring Convention) p.208-12 vol.2
Publisher: Printemps Convention, Paris, France
Publication Date: 1983 Country of Publication: France 2 vol. (319+432) pp.
Conference Date: 30 May-3 June 1983 Conference Location: Paris, France
Language: French Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: GALAAD is a software control and management system aimed primarily at the maintenance and delivery of software components. It comprises a software conceptual data base implemented using a librarian of text and binary, a data base management subsystem and a language adapted to software management, allowing access to all the data concerned and execution of the host operating system's own processors. The objects in the data base and the principal functions are presented by means of an example. This deals with the arrival of a software problem report, the correction of the source modules concerning the propagation of the updates , throughout the software product, with non-regression testing and creation of the new version . During the explanation of this example, the fundamental principles of GALAAD are shown, in particular the manner in which software evolutions in time and in space are managed. (0 Refs)
Subfile: C

34/7/52 (Item 10 from file: 6)
DIALOG(R)File 6:NTIS
(c) 2007 NTIS, Intl Cpyrght All Rights Res. All rts. reserv.

1715300 NTIS Accession Number: AD-A260 558/2
Design Management and Job Assignment System
(Technical rept. May 89-Mar 90)

Badr, S. M. ; Berzins, V.
Naval Postgraduate School, Monterey, CA. Dept. of Computer Science.
Corp. Source Codes: 019895006; 422721
Report No.: NPS-CS-92-020
Dec 92 30p
Languages: English
Journal Announcement: GRAI9311
Order this product from NTIS by: phone at 1-800-553-NTIS (U.S. customers); (703)605-6000 (other countries); fax at (703)321-8547; and email at orders@ntis.fedworld.gov. NTIS is located at 5285 Port Royal Road, Springfield, VA, 22161, USA.
NTIS Prices: PC A03/MF A01
Country of Publication: United States
This report introduces the basic approach taken in designing the design management and job assignment system (DMJAS) for CAPS93. This approach uses a model of software evolution (8) which defines how software changes , once it has been approved, will be applied to the software release (version) to produce another version of the software incorporating this change . This process is called an evolution step. The proposed system represents a management layer between the user interface (supporting two user classes, managers and designers) and the design database. The DMJAS controls the software evolution process in an incrementally evolving software system where the job steps to be scheduled are only partially known: time required, the set of sub-tasks for each step, and the input/output constraints between steps are all uncertain, and are all subject to change as steps are carried out. Models of design database and manager/designer interface are explained followed by the proposed system.... Software evolution , Job assignment, Design database, Evolution step, Software prototyping.

34/7/81 (Item 9 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)
(c) 2007 Elsevier Eng. Info. Inc. All rts. reserv.

06011792 E.I. Monthly No: EIM9101-001792
Title: Inverted approach to configuration management.
Author: Miller, David B.; Stockton, Robert G.; Krueger, Charles W.
Corporate Source: Carnegie Mellon Univ, Pittsburgh, PA, USA
Conference Title: Proceedings of the 2nd International Workshop on Software Configuration Management
Conference Location: Princeton, NJ, USA Conference Date: 19891024
Sponsor: ACM SIGSOFT; IEEE CS; Gesellschaft fuer Informatik
E.I. Conference No.: 13014
Source: Proc 2nd Int Workshop Software Configuration Manage. Publ by ACM, New York, NY, USA. p 1-4
Publication Year: 1989
ISBN: 0-89791-334-5
Language: English
Document Type: PA; (Conference Paper) Treatment: X; (Experimental)
Journal Announcement: 9101
Abstract: In the GANDALF project the authors are exploring the inverse approach by building version control on top of configuration management. The composition of variants of a system occurs at the lowest level. Each variant characterizes an instance of a complete system, such as a system variant to run under the X windowing system vs. a system variant to run under the NeWS windowing system. This approach emphasizes the composition of system as opposed to assembling system variants out of components organized with a version control system at the lowest level. The next higher level provides version control. This level maintains the evolution of the system as a set of revisions . As development proceeds, modifications of software components are tracked by the creation of additional revisions . Each revision contains a full set of variants in contrast to systems such as DSEE, which maintain revisions of

single components. At the top level, version control is embedded in a nested long-term transaction model to handle the issues of active software development by multiple users. This mechanism defines a recursive structure which provides encapsulated workspaces for software development and the ability to divide the development process into subtasks. All levels of the transaction hierarchy include the capabilities of the versioning system, providing the developer with a safe environment for experimental work. 4 Refs.

34/7/90 (Item 4 from file: 34)
DIALOG(R)File 34:SciSearch(R) Cited Ref Sci
(c) 2007 The Thomson Corp. All rts. reserv.

05668134 Genuine Article#: BH38E Number of References: 24
Title: Database schema evolution through the specification and maintenance of changes on entities and relationships
Author(s): Liu CT (REPRINT) ; Chrysanthis PK; Chang SK
Corporate Source: UNIV PITTSBURGH,DEPT COMP SCI/PITTSBURGH//PA/15260 (REPRINT)
, 1994, V881, P132-151
ISSN: 0302-9743 Publication date: 19940000
Publisher: SPRINGER-VERLAG BERLIN, HEIDELBERGER PLATZ 3, W-1000 BERLIN 33, GERMANYLECTURE NOTES IN COMPUTER SCIENCE
Series: LECTURE NOTES IN COMPUTER SCIENCE
Language: English Document Type: ARTICLE
Abstract: A flexible database system needs to support changes to its schema in order to facilitate the requirements of new applications and to support interoperability within a multidatabase system. In this paper, we present an approach to schema evolution through changes to the Entity-Relationship (ER) schema of a database. We enhance the graphical constructs used in ER diagrams, and develop EVER, an EVolutionary ER diagram for specifying the derivation relationships between schema versions , relationships among attributes, and the conditions for maintaining consistent views of programs. Algorithms are presented for mapping the EVER diagram into the underlying database and constructing database views for schema versions . Through the reconstruction of views after database reorganization, changes to an ER diagram can be made transparent to the application programs while all objects in the database remain accessible to the application programs.

34/7/101 (Item 2 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online
(c) 2007 ProQuest Info&Learning. All rts. reserv.

01683771 ORDER NO: AADMQ-35536
UNDERSTANDING SOFTWARE EVOLUTION USING THREE-DIMENSIONAL VISUALIZATION
Author: WANG, LI
Degree: M.SC.C.S.
Year: 1998
Corporate Source/Institution: THE UNIVERSITY OF NEW BRUNSWICK (CANADA) (0823)
Adviser: COLIN WARE
Source: VOLUME 37/03 of MASTERS ABSTRACTS.
PAGE 969. 133 PAGES
ISBN: 0-612-35536-5

Software evolution visualization is intended to help programmers and managers in visualizing and understanding the code changes that occur as a software system evolves. Limited research has been done on visualization of software evolution. Existing Version Control or Configuration Management tools have little capability to show changes graphically.

This dissertation research investigates a new approach to understanding software evolution using 3D graphics to visualize software changes . A prototype system, Code Version Visualization System (CVVS), has been developed to evaluate the possibility of visualizing changes of large production-sized software systems. CVVS uses a commercial software product, Cygnus Source-Navigator, to conduct code analysis and create Cygnus databases to store abstract information (about internal program structures and relationships between program components). Information about source code changes is retrieved from the Cygnus databases and merged into a single repository. Then, this information is converted to two types of data files designed specifically for a 3D software visualization tool, NV3D (NestedVision3D). NV3D was developed at Human Computer Interaction Lab at University of New Brunswick for visualizing a large code or network structure in 3D. E-NV3D, the viewer of CVVS, is a version of NV3D that has been enhanced as part of the thesis work to visualize software changes .

CVVS is evaluated with data from the telecommunications company Nortel and also with NV3D source code. CVVS is used to display version views with animated image changes and version comparisons with highlighted commonalties and differences between versions of a software system. The prototype system, CVVS, has clearly indicated the potential for software evolution visualization as part of a code management system.

34/7/110 (Item 11 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online
(c) 2007 ProQuest Info&Learning. All rts. reserv.

826157 ORDER NO: NOT AVAILABLE FROM UNIVERSITY MICROFILMS INT'L.
EDITING TECHNIQUES FOR MULTI-VERSION OBJECTS
Author: BROWN, DOUGLAS MARSHALL
Degree: PH.D.
Year: 1983
Corporate Source/Institution: UNIVERSITY OF SOUTHERN CALIFORNIA (0208)
Source: VOLUME 44/07-B OF DISSERTATION ABSTRACTS INTERNATIONAL.
PAGE 2195.

Software systems and other computer-maintained objects often evolve from single versions to families of versions. Family members may have differing characteristics, such as software with a requirement to function under different operating systems. A family of revisions, where each member consists of the state of the object at a given time, is an additional type of family. This dissertation presents techniques to support the maintenance of multi-version objects in the context of text editing. A single file stores the entire family, and the editor operates directly on the complete family. A detailed definition of editing operations on multi- version objects is presented. The techniques developed also incorporate merging and differencing operations, in order to build multi- version objects and combine independent changes to objects . A prototype system demonstrating the concepts developed has been implemented. The benefits gained from the approach lie in the visibility of the common and differing aspects of the object, the reliability with which the user can perform modifications across a family or within a particular member or members, and the simplicity with which the user can switch between the full-family and the single-member contexts.

(Copies available from Mmicrographics Department, Doheny Library, USC, Los Angeles, CA 90089.)

?

? t33/7/3-4

33/7/3 (Item 3 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

05806631 INSPEC Abstract Number: C9412-6120-012
Title: A linear-time scheme for version reconstruction
Author(s): Lin Yu; Rosenkrantz, D.J.
Author Affiliation: Dept. of Comput. Sci., State Univ. of New York, Albany, NY, USA
Journal: ACM Transactions on Programming Languages and Systems vol.16, no.3 p.775-97
Publication Date: May 1994 Country of Publication: USA
CODEN: ATPSDT ISSN: 0164-0925
U.S. Copyright Clearance Center Code: 0164-0925/94/0500-0775$03.50
Language: English Document Type: Journal Paper (JP)
Treatment: Practical (P); Theoretical (T)
Abstract: An efficient scheme to store and reconstruct versions of sequential files is presented. The reconstruction scheme involves building a data structure representing a complete version, and then successively modifying this data structure by applying a sequence of specially formatted differential files to it. Each application of a differential file produces a representation of an intermediate version , with the final data structure representing the requested version. The scheme uses a linked list to represent an intermediate version, instead of a sequential array, as is used traditionally. A new format for differential files specifying changes to this linked list data structure is presented. Algorithms are presented for using such a new format differential file to transform the representation of a version, and for reconstructing a requested version . Algorithms are also presented for generating the new format differential files, both for the case of a forward differential specifying how to transform the representation of an old version to the representation of a new version, and for the case of a reverse differential specifying how to transform the representation of a new version to the representation of an old version. (17 Refs)
Subfile: C

33/7/4 (Item 4 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2007 Institution of Electrical Engineers. All rts. reserv.

05013624 INSPEC Abstract Number: C91070325
Title: A linear-time scheme for version reconstruction
Author(s): Yu, L.; Rosenkrantz, D.J.
Author Affiliation: Dept. of Comput. Sci., State Univ. of New York, Albany, NY, USA
Conference Title: Algorithms and Data Structures. 2nd Workshop, WADS '91 Proceedings p.141-52
Editor(s): Dehne, F.; Sack, J.-R.; Santoro, N.
Publisher: Springer-Verlag, Berlin, Germany
Publication Date: 1991 Country of Publication: West Germany x+495 pp.
ISBN: 3 540 54343 0
Conference Sponsor: NSERC; ITRC; OCRI; TRIO; Carleton Univ.; Queen's Univ
Conference Date: 14-16 Aug. 1991 Conference Location: Ottawa, Ont., Canada
Language: English Document Type: Conference Paper (PA)
Treatment: Theoretical (T)
Abstract: An efficient scheme to store and reconstruct versions of sequential files is presented. The reconstruction scheme involves building a data structure representing a complete version, and then successively modifying this data structure by applying a sequence of specially formatted differential files to it. Each application of a differential

file produces a representation of an intermediate version , with the final data structure representing the requested version. The scheme uses a linked list to represent an intermediate version, instead of a sequential array, as is used traditionally. A new format for differential files specifying changes to this linked list data structure is presented. Algorithms are presented for using such a new format differential file to transform the representation for a version, and for reconstructing a requested version . Algorithms are also presented for generating the new format differential files, both for the case of a forward differential specifying how to transform the representation of an old version to the representation of a new version, and for the case of a reverse differential specifying how to transform the representation of a new version to the representation of an old version. (17 Refs)

Subfile: C

```
File 696:DIALOG Telecom. Newsletters 1995-2007/Mar 06
         (c) 2007 Dialog
File  15:ABI/Inform(R) 1971-2007/Mar 07
         (c) 2007 ProQuest Info&Learning
File  98:General Sci Abs 1984-2007/Mar
         (c) 2007 The HW Wilson Co.
File 484:Periodical Abs Plustext 1986-2007/Feb W3
         (c) 2007 ProQuest
File 813:PR Newswire 1987-1999/Apr 30
         (c) 1999 PR Newswire Association Inc
File 635:Business Dateline(R) 1985-2007/Mar 07
         (c) 2007 ProQuest Info&Learning
File 810:Business Wire 1986-1999/Feb 28
         (c) 1999 Business Wire
File 369:New Scientist 1994-2007/Nov W2
         (c) 2007 Reed Business Information Ltd.
File 370:Science 1996-1999/Jul W3
         (c) 1999 AAAS
File 624:McGraw-Hill Publications 1985-2007/Mar 07
         (c) 2007 McGraw-Hill Co. Inc
File 634:San Jose Mercury  Jun 1985-2007/Mar 04
         (c) 2007 San Jose Mercury News

Set     Items   Description
S1    1691595   MESSAGE? ? OR DOCUMENT? ? OR FILE OR FILES OR DATAFILE? OR
            SCHEMA? ? OR SCHEMATA? ? OR DATASTRUCTURE?
S2         63   TEXTMESSAGE? OR TEXTFILE? OR TEXTDATA OR COMPUTERFILE?
S3     813957   OBJECT? ? OR CONTENT? ?
S4    4147984   APPLICATION? ? OR PROGRAM? ? OR PROGRAMME OR PROGRAMMES OR
            APP OR APPS OR SOFTWARE? OR SOFT()WARE? ?
S5        113   SOURCECOD? OR PROGRAMCOD? OR PROGRAMMECOD?
S6      68451   PACKET? ? OR DATAPACKET? OR DATAGRAM? OR DATA()GRAM? ?
S7    4063602   UPDAT???? OR UPGRAD? OR UP()(GRAD???? OR DAT????) OR REVIS-
            ???? OR CHANG??? OR MODIFY? OR MODIFIE? ? OR MODIFICAT?
S8     655162   ADAPT??? OR ADAPTATION? OR RECONFIGUR? OR RE()CONFIGUR????-
            ?? OR ALTERR? OR ALTERATION? OR ALTER?? OR ALTERING
S9      73925   REFORMAT? OR REFRESH??? OR RE()(FORMAT???? OR FRESH???)
S10    367906   S7:S9(5N)(S1:S6 OR DATA OR DATUM? ?)
S11    658513   CHAIN???? OR INTERMEDIATE?
S12    339570   MAP OR MAPS OR MAPPED OR MAPPING? ? OR MAPED OR MAPING? ?
S13    547747   ITERAT???? OR PERMUTAT? OR EVOLV??? OR EVOLUTION?
S14   1759608   VERSION??? OR ADAPTATION? ? OR REVISION? ? OR EDITION? ? OR
             RELEASE OR RELEASES
S15     78972   S14(5N)(GENERAT???? OR BUILD??? OR BUILT OR CONSTRUCT???? -
            OR CREAT???? OR PRODUCE? ? OR PRODUCING OR PRODUCTION? ?)
S16     66781   S14(5N)(PROD? ? OR DERIV??? OR DERIVATION? ? OR PROPAGAT??-
            ?? OR DEFIN??? OR DEFINITION? OR ESTABLISH? OR DEVELOP?????)
S17     14054   S14(5N)(FORM OR FORMS OR FORMED OR FORMING OR FORMATION? ?)
S18      4173   S15:S17(S)(S11 OR S13)
S19       209   S18(S)S10
S20       124   S19/1999:2007
S21        85   S19 NOT S20
S22        75   RD  (unique items)

 22/3,K/23     (Item 19 from file: 15)
DIALOG(R)File  15:ABI/Inform(R)
(c) 2007 ProQuest Info&Learning. All rts. reserv.

00952038  96-01431
Database requirements for CIM applications
Kappel, Gerti; Vieweg, Stefan
Integrated Manufacturing Systems  v5n4,5  PP: 48-63  1994
ISSN: 0957-6061  JRNL CODE: ING
```

WORD COUNT: 11707

...TEXT: goals, which have to be satisfied. Next to version management, evolving applications such as CIM applications must cope with changing requirements, which imply changes in the description of the data in the database known as schema evolution and schema versioning. Both version management and schema evolution /versioning have to cope with changes either of data values or of data descriptions. These tasks are commonly referred to as change management[37...

...branching versions. Revisions represent improvements of a specified version. Alternatives describe different ways of manipulating objects affecting the same result. Revisions and alternatives should be combined in any order. Together the representations form a version tree where a node is linked to its parent node via a derivedFrom-relationship (= versionOf...office information systems may also require version management. An example of the former application are versions of production plans, an example of the latter is co-operative editing of documents .

Let us now turn to changes in the description of the data in the database, to schema evolution and schema versioning. The database schema describes the structure...

22/3,K/40 (Item 36 from file: 15)
DIALOG(R)File 15:ABI/Inform(R)
(c) 2007 ProQuest Info&Learning. All rts. reserv.

00522300 90-48057
The Evolution Support Environment System
Ramamoorthy, C. V.; Usuda, Yutaka; Prakash, Atul; Tsai, W. T.
IEEE Transactions on Software Engineering v16n11 PP: 1225-1234 Nov 1990
ISSN: 0098-5589 JRNL CODE: ISO

ABSTRACT: As software systems evolve over time, modifications require understanding relations among components of the software system. A framework is provided by the Evolution Support Environment (ESE) system for capturing and making available semantic information about software components of an evolving software system. The ESE system is designed to provide integrated support for management of software...

...While life-cycle management allows traceability among specifications, design, code, and test cases during software development , version control allows specific versions of software objects and their associated objects, such as specifications and test cases, to be...

```
File 347:JAPIO Dec 1976-2006/Nov(Updated 070228)
        (c) 2007 JPO & JAPIO
File 348:EUROPEAN PATENTS 1978-2007/ 200708
        (c) 2007 European Patent Office
File 349:PCT FULLTEXT 1979-2007/UB=20070301UT=20070222
        (c) 2007 WIPO/Thomson
File 350:Derwent WPIX 1963-2006/UD=200716
        (c) 2007 The Thomson Corporation

Set     Items   Description
S1         18   AU='MOUTON J':AU='MOUTON J A'
S2          2   AU='MOUTON JEROME ANTHONY JR':AU='MOUTON JEROME ANTHONY JR
             3439 CORK OAK WAY PAL'
S3          0   AU='VAN TRUONG'
S4          0   AU='VANTRUONG'
S5          2   AU='BUCKLEY W'
S6          7   AU='BUCKLEY W P'
S7          1   AU='BUCKLEY WILLIAM'
S8          8   AU='BUCKLEY WILLIAM P':AU='BUCKLEY WILLIAM PETER'
S9         32   S1:S8
S10       460   (UPGRAD? OR UP()GRAD???)(5N)(DATABASE? OR DATA()BASE? ?)
S11         0   S9 AND S10
S12    779454   CHAIN???
S13         1   S9 AND S12
S14     40757   UPGRAD? OR UP()GRAD???
S15         0   S9 AND S14
? t13/5

 13/5/1     (Item 1 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS
(c) 2007 European Patent Office. All rts. reserv.

00405124
Method for synchronizing the dispatching of tasks among multitasking
    operating systems.
Verfahren      fur      die     Tasksynchronisierungszuweisung      zwischen
    Multiaufgabenbetriebssystemen.
Methode de synchronisation de l'attribution des taches entre des systemes
    operationnels multitaches.
PATENT ASSIGNEE:
  International Business Machines Corporation, (200120), Old Orchard Road,
    Armonk, N.Y. 10504, (US), (applicant designated states: DE;FR;GB)
INVENTOR:
  Goldberg, Steven Hirsh, 30 Pear Court, Hillsborough, CA 94010, (US)
  Holten, Gerald Wayne, 4398 Hilton Avenue, San Jose, Ca 95130, (US)
   Mouton, Jerome Anthony, Jr. , 3439 Cork Oak Way, Palo Alto, CA 94303,
    (US)
LEGAL REPRESENTATIVE:
  Moss, Robert Douglas et al (34141), IBM United Kingdom Limited
    Intellectual Property Department Hursley Park, Winchester Hampshire
    SO21 2JN, (GB)
PATENT (CC, No, Kind, Date):  EP 381655  A2  900808 (Basic)
                              EP 381655  A3  921202
APPLICATION (CC, No, Date):   EP 90850040 900130;
PRIORITY (CC, No, Date): US 304740 890131
DESIGNATED STATES: DE; FR; GB
INTERNATIONAL PATENT CLASS (V7): G06F-009/46;
CITED REFERENCES (EP A):
  IBM TECHNICAL DISCLOSURE BULLETIN vol. 30, no. 5, October 1987, NEW YORK,
    US pages 373 - 374 'Selective Dispatch of Queued Tasks'
  PETERSON AND SILBERSCHATZ: 'Operating System Concepts' 1983 ,
    ADDISON-WESLEY PUBLISHING CO.
  PROCEEDINGS OF THE SIXTH ANNUAL INTERNATIONAL PHOENIX CONFERENCE ON
    COMPUTERS AND COMMUNICATIONS, 25 - 27 February  1987, SCOTTSDALE,
```

ARIZONA , US pages 475 - 481 L. CIMINIERA ET AL. : 'Mechanisms for Communication Resource Management in ISO/OSI Layers';

ABSTRACT EP 381655 A2

A method for synchronizing the dispatching of tasks from a CPU-based first multitasking operating system (OS) with threads of function calls opportunistically dispatched from a CPU-based second multitasking operating system. The second OS includes a set of callable resources. In the method, a task becomes bound to a thread for the duration of that thread's ownership of a callable resource from the second OS. Also, a thread becomes available on a work queue of threads for binding to a task only if the thread owns exactly one resource. After execution, the function is eliminated from the thread and ownership of that resource is relinquished and passed to the next thread queued on that resource. A task can remain bound to the same thread if there are no other threads asserting ownership to the next resource being called by the original thread. With contention, however, the task relinquishes the thread and becomes bound to another thread on the work queue. (see image in original document)

ABSTRACT WORD COUNT: 168

LEGAL STATUS (Type, Pub Date, Kind, Text):

Application:	900808	A2	Published application (A1with Search Report ;A2without Search Report)
Examination:	910109	A2	Date of filing of request for examination: 901113
Search Report:	921202	A3	Separate publication of the European or International search report
Change:	921230	A2	Representative (change)
Withdrawal:	960110	A2	Date on which the European patent application was withdrawn: 951120

LANGUAGE (Publication,Procedural,Application): English; English; English

FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	EPABF1	857
SPEC A	(English)	EPABF1	4242
Total word count - document A			5099
Total word count - document B			0
Total word count - documents A + B			5099

09/295,690

File 347:JAPIO Dec 1976-2006/Nov(Updated 070228)
(c) 2007 JPO & JAPIO
File 350:Derwent WPIX 1963-2006/UD=200716
(c) 2007 The Thomson Corporation

Set	Items	Description
S1	453955	MESSAGE? ? OR DOCUMENT? ? OR FILE OR FILES OR DATAFILE? OR SCHEMA? ? OR SCHEMATA? ? OR DATASTRUCTURE?
S2	8	TEXTMESSAGE? OR TEXTFILE? OR TEXTDATA OR COMPUTERFILE?
S3	1533174	OBJECT? ? OR CONTENT? ?
S4	1731769	APPLICATION? ? OR PROGRAM? ? OR PROGRAMME OR PROGRAMMES OR APP OR APPS OR SOFTWARE? OR SOFT()WARE? ?
S5	14	SOURCECOD? OR PROGRAMCOD? OR PROGRAMMECOD?
S6	106313	PACKET? ? OR DATAPACKET? OR DATAGRAM? OR DATA()GRAM? ?
S7	2125038	UPDAT???? OR UPGRAD? OR UP()(GRAD???? OR DAT????) OR REVIS-???? OR CHANG??? OR MODIFY? OR MODIFIE? ? OR MODIFICAT?
S8	846874	ADAPT??? OR ADAPTATION? OR RECONFIGUR? OR RE()CONFIGUR????-?? OR ALTERR? OR ALTERATION? OR ALTER?? OR ALTERING
S9	27560	REFORMAT? OR REFRESH??? OR RE()(FORMAT???? OR FRESH???)
S10	242009	S7:S9(5N)(S1:S6 OR DATA OR DATUM? ?)
S11	929025	CHAIN???? OR INTERMEDIATE?
S12	110543	MAP OR MAPS OR MAPPED OR MAPPING? ? OR MAPED OR MAPING? ?
S13	41711	ITERAT???? OR PERMUTAT? OR EVOLV??? OR EVOLUTION?
S14	518861	VERSION??? OR ADAPTATION? ? OR REVISION? ? OR EDITION? ? OR RELEASE OR RELEASES
S15	18311	S14(5N)(GENERAT???? OR BUILD??? OR BUILT OR CONSTRUCT???? - OR CREAT???? OR PRODUCE? ? OR PRODUCING OR PRODUCTION? ?)
S16	7174	S14(5N)(PROD? ? OR DERIV??? OR DERIVATION? ? OR PROPAGAT??-?? OR DEFIN??? OR DEFINITION? OR ESTABLISH? OR DEVELOP?????)
S17	22339	S14(5N)(FORM OR FORMS OR FORMED OR FORMING OR FORMATION? ?)
S18	3396	S15:S17 AND S11:S13
S19	180	S18 AND S10
S20	68	S19 AND AC=US/PR AND AY=(1963:1999)/PR
S21	79	S19 AND AC=US AND AY=1963:1999
S22	79	S19 AND AC=US AND AY=(1963:1999)/PR
S23	67	S19 AND PY=1963:1999
S24	94	S20:S23

? t24/9/4,6

24/9/4 (Item 4 from file: 347)
DIALOG(R)File 347:JAPIO
(c) 2007 JPO & JAPIO. All rts. reserv.

05182184 **Image available**
PROGRAM TRANSFER METHOD, SYSTEM AND DEVICE, FOR PROGRAM DEVELOPMENT, AND PROGRAM EXECUTION DEVICE

PUB. NO.: 08-137684 [JP 8137684 A]
PUBLISHED: May 31, 1996 (19960531)
INVENTOR(s): TANAKA TOSHIYUKI
APPLICANT(s): SHARP CORP [000504] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 06-278945 [JP 94278945]
FILED: November 14, 1994 (19941114)
INTL CLASS: [6] G06F-009/06; G06F-009/06
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)

ABSTRACT

PURPOSE: To reduce the transfer time by transferring only an object file subject to revision to a target device again and linking the object file with other object file having been transferred in an order.

CONSTITUTION: Intermediate object programs being the result of compiling

files 1 to 5 are intermediate files 70a, 72a, 74a, 76a and 78a. Then unuse areas 70b to 78b are provided in advance to the object files 70a to 78a. When, e.g. the file 72a is corrected, only the object program after correction is transferred to the program execution device. Then the transferred object file is linked in the order that the object file has the same address in the execution form program as that of the address before the revision . Thus, the executionable form program after the revision is generated in the program execution device by having only to link the revised and transferred object file with the other object files having been transferred in an order.

24/9/6 (Item 6 from file: 347)
DIALOG(R)File 347:JAPIO
(c) 2007 JPO & JAPIO. All rts. reserv.

04000047 **Image available**
DOCUMENT MANAGEMENT SYSTEM

PUB. NO.: 04-365147 [JP 4365147 A]
PUBLISHED: December 17, 1992 (19921217)
INVENTOR(s): SUMIYA KIYOO
APPLICANT(s): TOSHIBA CORP [000307] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 03-141755 [JP 91141755]
FILED: June 13, 1991 (19910613)
INTL CLASS: [5] G06F-012/00; G06F-015/20
JAPIO CLASS: 45.2 (INFORMATION PROCESSING -- Memory Units); 45.4 (INFORMATION PROCESSING -- Computer Applications)
JOURNAL: Section: P, Section No. 1535, Vol. 17, No. 242, Pg. 130, May 14, 1993 (19930514)

ABSTRACT
PURPOSE: To leave the rough copy version (rough copy document), where the change record for version change are entered on the copy of the version before the change, in a form easy to understand as an official document.
CONSTITUTION: The document of a version (m) as the version change object is copied to another area of a document data storage part 21 with another document name by a documentation part 23 so that the same copy is handled as an intermediate version m, j (J=1, 2...), and a pair of this version m, j and the document name are registered as one entry in the document ledger of a document ledger storage part 22 by the document preparing part 23. The change record for version up is added to the copy by a DTP function part 27, and this copy is formally registered as the intermediate version m, j by a register part 28, and this intermediate version m, j is referred to generate a next version (n) (=m+1) from the version (m).
? t24/12-14,17,20,32,40,42,47
>>>'-' not allowed as format type
? t24/69,k/12-14,17,20,32,40,42,47

24/69,K/12 (Item 4 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0013938315 - Drawing available
WPI ACC NO: 2004-118522/200412
XRPX Acc No: N2004-094700
Collaborative model for distributed software system, has merge sub-model to resolve conflicts due to change in representation of object in construct and to propagate notification of conflict result
Patent Assignee: QWEST COMMUNICATIONS INT INC (QWES-N)
Inventor: HABERMEHL K D; HURLEY W D
Patent Family (1 patents, 1 countries)
Patent Application

Number	Kind	Date	Number	Kind	Date	Update	
US 6678882	B1	20040113	US 1999345044	A	19990630	200412	B

Priority Applications (no., kind, date): US 1999345044 A 19990630

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
US 6678882	B1	EN	25	9	

Alerting Abstract US B1

NOVELTY - A change sub-model (16) defines changes in instances and types of construct defined by type model (22). A replication sub-model (18) defines different representation of objects of construct. A synchronization sub-model (12) optimizes software re-use and reports change in replication sub-model. A merge sub-model (14) resolves conflicts defined in collection of changes.

DESCRIPTION - A distribution model (20) defines physical transportation of objects.

USE - Model for supporting construction of collaborative software system in which multiple agents collaborate and communicate by sharing objects e.g. world wide web as wireless network.

ADVANTAGE - Resolving conflicts of change in object representation avoids overhead in programming and provides more efficient distribution of shared object.

DESCRIPTION OF DRAWINGS - The figure shows the block diagram illustrating the sub-models of collaboration model.

10 collaboration model
12 synchronization sub-model
14 merge sub-model
16 change model
18 replication model
20 distribution model
22 type model

Title Terms/Index Terms/Additional Words: MODEL; DISTRIBUTE; SOFTWARE; SYSTEM; MERGE; SUB; RESOLUTION; CHANGE; REPRESENT; OBJECT; CONSTRUCTION; PROPAGATE; NOTIFICATION; CONFLICT; RESULT

Class Codes
International Classification (Main): G06F-009/44

File Segment: EPI;
DWPI Class: T01
Manual Codes (EPI/S-X): T01-F07; T01-J20A; T01-N01A2D

Collaborative model for distributed software system, has merge sub-model to resolve conflicts due to change in representation of object in construct and to propagate notification of conflict result

Alerting Abstract ...model (18) defines different representation of objects of construct. A synchronization sub-model (12) optimizes software re-use and reports change in replication sub-model. A merge sub-model (14) resolves conflicts defined in collection of...

...ADVANTAGE - Resolving conflicts of change in object representation avoids overhead in programming and provides more efficient distribution of shared object...

Original Publication Data by Authority

Original Abstracts:
A collaboration model is provided for constructing evolutionary collaborative applications deployed in network environments, such as the World Wide Web, or wireless environments, which supports introduction of...

...conflicts whenever possible. A change submodel is operative to define allowable changes for shared objects. A replication model is arranged to provide replication functionality, and define how multiple versions of an object are presented. A distribution submodel provides distribution functionality, and a type submodel provides a programming language...
Claims:
...arranged to define a manner in which multiple versions of objects are represented;a synchronization submodel arranged to collect and report changes in the replication submodel and to optimize software reuse in isolation of potential changes;a merge submodel arranged to initiate collection of changes , identify conflicts, resolve conflicts, and propagate notification of whether changes were accepted, rejected, or a conflict could not be resolved; and...
Basic Derwent Week: 200412

24/69,K/13 (Item 5 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0013689561 - Drawing available
WPI ACC NO: 2003-786428/200374
XRPX Acc No: N2003-630180
Computerized heterogeneous program translation method involves resolving logical linkages to absolute addresses, corresponding to code blocks, to output specific instructions to create new version of component
Patent Assignee: MICROSOFT CORP (MICT)
Inventor: SRIVASTAVA A; VO H H
Patent Family (1 patents, 1 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update
US 6609248	B1	20030819	US 1999343287	A	19990630	200374 B

Priority Applications (no., kind, date): US 1999343287 A 19990630

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
US 6609248	B1	EN	22	7	

Alerting Abstract US B1
NOVELTY - Logical linkages of code blocks in intermediate representation (220) of components in heterogeneous program are created and are resolved to absolute addresses assigned to respective code blocks. A platform specific executable instruction for each instruction represented in the intermediate component representation is written in the buffer to generate an output file for creating a new component version .
DESCRIPTION - INDEPENDENT CLAIMS are also included for the following:

1.computer readable medium storing translation program,

2.heterogeneous program translating system.

USE - For translating heterogeneous program into different architectures, in personal computer, handheld device, multiprocessor system, microprocessor-based or programmable consumer electronics, network PC, minicomputer, mainframe computer and distributed computer system.
ADVANTAGE - Capable of efficiently converting specific portions of program to optimize the execution of program, for producing compact code. Also enables to change order of code blocks within the component while maintaining and optimizing the control flow and generates a more efficient heterogeneous program from homogeneous program.
DESCRIPTION OF DRAWINGS - The figure shows the functional block diagram of code translation system.

200 code translation system
210 reader
220 intermediate representation
230 transformation module
240 writer

Title Terms/Index Terms/Additional Words: COMPUTER; HETEROGENEOUS; PROGRAM; TRANSLATION; METHOD; RESOLUTION; LOGIC; LINK; ABSOLUTE; ADDRESS; CORRESPOND; CODE; BLOCK; OUTPUT; SPECIFIC; INSTRUCTION; NEW; VERSION; COMPONENT

Class Codes
International Classification (Main): G06F-009/45

File Segment: EPI;
DWPI Class: T01
Manual Codes (EPI/S-X): T01-F05A; T01-J14; T01-J20A; T01-S03

...resolving logical linkages to absolute addresses, corresponding to code blocks, to output specific instructions to create new version of component

Alerting Abstract ...NOVELTY - Logical linkages of code blocks in intermediate representation (220) of components in heterogeneous program are created and are resolved to absolute addresses...

...to respective code blocks. A platform specific executable instruction for each instruction represented in the intermediate component representation is written in the buffer to generate an output file for creating a new component version . ...220 intermediate representation

Original Publication Data by Authority

Original Abstracts:
...translator provides for cross module representations of components within a heterogeneous program by translating modifying a platform-neutral intermediate representation (IR) of the program into platform-specific instructions for different architectures. The intermediate representation is hierarchy of base elements that correspond to instructions, code blocks, procedures and components within the program. Blocks...

...were originally written for one architecture can be translated from the intermediate representation into platform- specific instructions for a different architecture. The output translator provides any necessary code to interface contiguous...
Claims:
...obtaining a platform-neutral intermediate representation of a component in the heterogeneous program;creating logical linkages among a plurality of code blocks in the intermediate representation of the component to establish a control flow through the component; assigning an absolute address within an address space for the component to each of the plurality ...

...component into a buffer; andwriting the buffer to an output file to create a new version of the component. Basic Derwent Week: 200374

24/69,K/14 (Item 6 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0013624356 - Drawing available
WPI ACC NO: 2003-719872/200368
XRPX Acc No: N2003-575434
Data object modification representing method for updating files stored in read only media, involves generating update information corresponding to difference between data objects in different versions
Patent Assignee: IORA LTD (IORA-N)
Inventor: COLLINS B J; DRAPER S P W; FALLS P T
Patent Family (1 patents, 1 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update
US 6604236	B1	20030805	US 199891291	P	19980630	200368 B
			US 1999340411	A	19990628	

Priority Applications (no., kind, date): US 199891291 P 19980630; US 1999340411 A 19990628

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
US 6604236	B1	EN	23	5	Related to Provisional US 199891291

Alerting Abstract US B1

NOVELTY - The meta data for data objects in different versions, are stored in basis and delta directory entry tables respectively. The difference between the data objects is identified by comparing their meta objects . An update information corresponding to identified result is generated. The update information is used to generate one version of data object from the another version of data object.

DESCRIPTION - An INDEPENDENT CLAIM is also included for method of modifying set of data objects with data from another set of data objects.

USE - For representing modification of set of data objects for updating content of copy of original files stored in read only media e.g. compact disk-read only memory (CD-ROM).

ADVANTAGE - The content of the original files are updated easily without generating a copy of every file. Reduces the amount of data to be stored in the delta directory from delta modification file .

DESCRIPTION OF DRAWINGS - The figure shows the flowchart explaining the data object modification representing method.

Title Terms/Index Terms/Additional Words: DATA; OBJECT; MODIFIED; REPRESENT ; METHOD; UPDATE; FILE; STORAGE; READ; MEDIUM; GENERATE; INFORMATION; CORRESPOND; DIFFER; VERSION

Class Codes
International Classification (Main): G06F-009/45

File Segment: EPI;
DWPI Class: T01
Manual Codes (EPI/S-X): T01-F05E

Data object modification representing method for updating files stored in read only media, involves generating update information corresponding to difference between data objects in different versions

Original Titles:
System and method for generating file updates for files stored on read-only media

Alerting Abstract ...entry tables respectively. The difference between the data objects is identified by comparing their meta objects . An update information corresponding to identified result is generated. The update information is used to generate one version of data object from the another version of data object.DESCRIPTION - An INDEPENDENT CLAIM is also included for method of modifying set of data objects with data

from another set of data objects...

...USE - For representing modification of set of data objects for updating content of copy of original files stored in read only media e.g. compact disk-read only memory (CD-ROM...

...ADVANTAGE - The content of the original files are updated easily without generating a copy of every file. Reduces the amount of data to be stored in the delta directory from delta modification file .
...

...DESCRIPTION OF DRAWINGS - The figure shows the flowchart explaining the data object modification representing method.

Original Publication Data by Authority

Original Abstracts:
A method and system of the present invention generates a representation of a new version of an original file system with reference to the original file system and the new version of the file...
...the amount of data to be stored in the delta directory map file, delta modification data block file, and delta look up table generated for the data portions unique to the newest version of an original file...

...identify the location of data portions that may be used to generate the newest version of the file system . The data portions may be located in a file in the original file system, a delta modification data block file in a previous version of the original file system or a delta modification data block file for the newest version of the original file system.The method of the present invention is performed by generating
...

...the data content of an original file system, generating a file of modification data blocks that may be used to modify the data content of the original file system, and generating a delta look up table that identifies the location of the data blocks used to represent the
...

...delta look up table and the file of modification data blocks may be stored for delivery to a computer on which a copy of the original file system is stored. The delta look up table and the file of modification data blocks are then used by the computer system on which a copy of the original file system is stored to provide the data content...

...invention may be used to generate data for updating the content of a copy of the original file system without having to generate a copy of every file and data block for the new content...
Claims:
...is claimed is:1. A method for representing modifications to a set of data objects comprising :generating a baseline of a first version of a set of data objects;identifying the differences between a second version of the set of data objects and the first version of...

...a new data object or a modified data object;comparing meta-data for data objects in said baseline to meta - data for data objects in said second version of said data objects to determine whether one of...

...generating update information corresponding to the identified differences, said update information including references to segmented portions of said baseline that correspond to said identified differences regardless of location of said segmented portions in said baseline; and
Basic Derwent Week: 200368

24/69,K/17 (Item 9 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0013118104 - Drawing available
WPI ACC NO: 2003-199812/200319
Related WPI Acc No: 2003-391938; 2003-446836; 2004-255886
XRPX Acc No: N2003-159016
Database objects schema evolution method for front office automation application , involves transmitting changes made in master table of headquarter's site to sales officer site for updating replicated tables in that site
Patent Assignee: DEMERS A J (DEME-I); ELSBERND C (ELSB-I); STAMOS J W (STAM-I); WONG L (WONG-I); ORACLE INT CORP (ORAC-N)
Inventor: DEMERS A J; ELSBERND C; STAMOS J W; WONG L
Patent Family (2 patents, 1 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update	
US 20020174142	A1	20021121	US 199886985	P	19980528	200319	B
			US 1999321594	A	19990528		
US 7162689	B2	20070109	US 1999321594	A	19990528	200705	E

Priority Applications (no., kind, date): US 199886985 P 19980528; US 1999321594 A 19990528
Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
US 20020174142	A1	EN	14	6	Related to Provisional US 199886985

Alerting Abstract US A1
NOVELTY - Replicated copies (411,412) of a master table provided in a company headquarter's site (400) are stored in sales officer sites (402,404), respectively. When changes are made in the master table, they are transmitted to the sales officer site, for updating the tables in those sites.
DESCRIPTION - An INDEPENDENT CLAIM is included for method of modifying master table.
USE - For implementing schema evolution database objects in large-scale distributed system such as front office automation environment.
ADVANTAGE - By transmitting the changes made in the master table to the replicated tables provided in different sites, there is no need to keep same object group at all the sites. Thus, the quiescence for front office automation deployments can be performed easily.
DESCRIPTION OF DRAWINGS - The figure shows a schematic view of the distributed database system during the schema evolution process.
400 Company headquarter's site
402,404 Sales officer sites
411,412 Replicated copies

Title Terms/Index Terms/Additional Words: DATABASE; OBJECT; EVOLVE ; METHOD; FRONT; OFFICE; AUTOMATIC; APPLY; TRANSMIT; CHANGE; MADE; MASTER; TABLE; SITE; SALE; UPDATE; REPLICA

Class Codes
International Classification (+ Attributes)

IPC + Level	Value	Position	Status	Version
G06F-0017/30	A	I	R	20060101
G06F-0017/00	A	I F	B	20060101
G06F-0017/30	C	I	R	20060101

File Segment: EPI;
DWPI Class: T01; W01
Manual Codes (EPI/S-X): T01-J05A2; T01-J05B4A; W01-A06E2B

Database objects schema evolution method for front office automation application , involves transmitting changes made in master table of headquarter's site to sales officer site for updating replicated...

Original Titles:
SCHEMA EVOLUTION IN REPLICATION...

...Schema evolution in replication

Alerting Abstract ...USE - For implementing schema evolution database objects in large-scale distributed system such as front office automation environment...

...DRAWINGS - The figure shows a schematic view of the distributed database system during the schema evolution process...

Title Terms.../Index Terms/Additional Words: EVOLVE ;

Original Publication Data by Authority

Original Abstracts:
Schema evolution involves defining flavored object groups. Specifically, related collections of replicated tables and other database objects...

...the master tables. In one embodiment, when one site in a distributed database system propagates changes to a replicated database object , the current flavor for the site is also transmitted, so that the destination site can...

...Schema evolution involves defining flavored object groups. Specifically, related collections of replicated tables and other database objects...

...the master tables. In one embodiment, when one site in a distributed database system propagates changes to a replicated database object , the current flavor for the site is also transmitted, so that the destination site can...
Claims:
...at the second site, from a first site maintaining a first version of the table, change data that identifies changes made to the first version of the table; andupdating, at the second site, the second version of the table based on the change data to create an updated second version of the table,wherein at least one of the first version of the table and...
Basic Derwent Week: 200319

24/69,K/20 (Item 12 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0012348584 - Drawing available
WPI ACC NO: 2002-290939/200233
XRPX Acc No: N2002-227107
Software updating method in computer system, involves producing final hub version using hub incremental update and original hub version without using any intermediate versions
Patent Assignee: SYMANTEC CORP (SYMA-N)
Inventor: NACHENBERG C S; SOBEL W E
Patent Family (1 patents, 1 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update
US 6349311	B1	20020219	US 1999241794	A	19990201	200233 B

Priority Applications (no., kind, date): US 1999241794 A 19990201

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
US 6349311	B1	EN	7	3	

Alerting Abstract US B1

NOVELTY - An original hub version (1.0) is produced using an original back- up date and an original software version. A final hub version (3.0) is produced using a hub incremental update (110) and the original hub version without using any intermediate versions . A final version is produced in response to the final hub version, using a final incremental update (112) and the final hub version.

USE - For updating computer software applications .

ADVANTAGE - Permits updating from a small number of incremental updates, without requiring the storage of a large number of incremental update files on the server. Hence requires a relatively small amount of local storage space.

DESCRIPTION OF DRAWINGS - The figure shows an illustration of a sequence of file versions and the incremental updates .

1.0 Original hub version

3.0 Final hub version

110 Hub incremental update

112 Final incremental update

Title Terms/Index Terms/Additional Words: SOFTWARE; UPDATE; METHOD; COMPUTER; SYSTEM; PRODUCE; FINAL; HUB; VERSION; INCREMENT; ORIGINAL; INTERMEDIATE

Class Codes

International Classification (Main): G06F-012/00

(Additional/Secondary): G06F-017/30

File Segment: EPI;

DWPI Class: T01

Manual Codes (EPI/S-X): T01-J20B2

Software updating method in computer system, involves producing final hub version using hub incremental update and original hub version without using any intermediate versions

Alerting Abstract ...NOVELTY - An original hub version (1.0) is produced using an original back- up date and an original software version. A final hub version (3.0) is produced using a hub incremental update (110) and the original hub version without using any intermediate versions . A final version is produced in response to the final hub version, using a final incremental update (112) and the...

USE - For updating computer software applications .

...

...small number of incremental updates, without requiring the storage of a large number of incremental update files on the server. Hence requires a relatively small amount of local storage space...

...DESCRIPTION OF DRAWINGS - The figure shows an illustration of a sequence of file versions and the incremental updates .

Title Terms.../Index Terms/Additional Words: INTERMEDIATE

Original Publication Data by Authority

Original Abstracts:

...incremental update (112) which provides the information necessary to construct the file of the second version (3.2) from a file of the first version (3.2). In order to allow for future access to the first...
...back-update file (206) is created. The back-update file (206) provides the information necessary to construct a file of the first state (3. 0) from a file of the second state (3.2).
Claims:
A method for updating a computer readable file from an original state to a final state, the original state residing on a computer system, the...

...being the final hub state; wherein at least one intermediate state is between the original hub state and the final hub state:using a hub incremental update and the original hub...

...a final hub state without using any intermediate states; andresponsive to the final hub state not being the final state:using a final incremental update and the computer readable file of the final hub state to produce a final state;retaining the final state;producing a final back-update for producing the final hub state from the...
Basic Derwent Week: 200233

24/69,K/32 (Item 24 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0009971114 - Drawing available
WPI ACC NO: 2000-273625/200024
XRPX Acc No: N2000-205127
Method for representing modifications to set of data objects to update information stored on ROM media by storing update information that may be retrieved and used to generate second version of set of data objects
Patent Assignee: IORA LTD (IORA-N)
Inventor: COLLINS B J; DRAPER S P W; FALLS P T
Patent Family (1 patents, 25 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update
EP 994425	A2	20000419	EP 1999112710	A	19990630	200024 B

Priority Applications (no., kind, date): US 199891291 A 19980630

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
EP 994425	A2	EN	17	5	

Regional Designated States,Original: AL AT BE CH CY DE DK ES FI FR GB GR IE IT LI LT LU LV MC MK NL PT RO SE SI

Alerting Abstract EP A2
NOVELTY - An update information includes references to segmented portions of the baseline that correspond to the identified differences regardless of location of the segmented portions in the baseline. The update information is stored so as the update information may be retrieved and used to generate the second version of the set of data objects from the first version of the set of data objects.
DESCRIPTION - An INDEPENDENT CLAIM is included for:

1.a method for providing data associated with a data object with an updated set of data object stored on computer

USE - In file updating methods for files stored on read -only media.
ADVANTAGE - Allows to update information stored on read-only media

without having to produce new read-only media containing the program or data updates for distribution to a company ' s customers or remote personnel.
DESCRIPTION OF DRAWINGS - The drawing is a flowchart of exemplary process that generates a delta directory map file for the new version of the original file system from the delta directory entry meta table generated by look-up table file.
Title Terms/Index Terms/Additional Words: METHOD; REPRESENT; MODIFIED; SET; DATA; OBJECT; UPDATE; INFORMATION; STORAGE; ROM; MEDIUM; RETRIEVAL; GENERATE; SECOND; VERSION

Class Codes
International Classification (Main): G06F-017/30

File Segment: EPI;
DWPI Class: T01
Manual Codes (EPI/S-X): T01-F01B; T01-J05B; T01-J20B2

Method for representing modifications to set of data objects to update information stored on ROM media by storing update information that may be retrieved and used to generate second version of set of data objects

Original Titles:
...System and method for generating file updates for files stored on read-only media...

Alerting Abstract ...update information is stored so as the update information may be retrieved and used to generate the second version of the set of data objects from the first version of the set of data...
...a method for providing data associated with a data object with an updated set of data object stored on computer
...

...USE - In file updating methods for files stored on read -only media.
...

...having to produce new read-only media containing the program or data updates for distribution to a company ' s customers or remote personnel
...

...flowchart of exemplary process that generates a delta directory map file for the new version of the original file system from the delta directory entry meta table generated by look-up

Original Publication Data by Authority

Original Abstracts:
This invention relates to file updating methods, and more particularly to file updating methods for files stored on Read- Only media. To represent modifications to a set of data objects the method according to the present invention comprises generating a baseline of a first version of a set of data objects; identifying the differences between a second version of the set of data objects and the first version of the set of the data objects; generating update information corresponding to the identified differences, said update information including references to segmented portions of said baseline that correspond...

...information whereby the update information may be retrieved and used to generate the second version of the said of data objects from the first version of the said of the data objects.
Claims:
A method for representing modifications to a set of data objects

comprising : generating a baseline of a first version of a set of data objects;identifying the differences between a second version of the set of data objects and the first version of the set of data objects;generating update information corresponding to the identified differences, said update information including references to segmented portions of said baseline that correspond to said identified differences...

...be retrieved and used to generate the second version of the set of data objects from the first version of the set of data objects.
...Basic Derwent Week: EP 1999112710

24/69,K/40 (Item 32 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0009657617 - Drawing available
WPI ACC NO: 1999-610509/ 199952
XRPX Acc No: N1999-449825
Class version naming and mapping system in object oriented computer system
Patent Assignee: INT BUSINESS MACHINES CORP (IBMC)
Inventor: GERARD S N; MUNROE S J
Patent Family (1 patents, 1 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update	
US 5974428	A	19991026	US 1997920456	A	19970829	199952	B

Priority Applications (no., kind, date): US 1997920456 A 19970829

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
US 5974428	A	EN	11	4	

Alerting Abstract US A
NOVELTY - When a request is made to load a default class, a version loader object checks a version mapping mechanism to determine a preferred version of default class. The preferred version of default class determined is then loaded by version loader object.
DESCRIPTION - A memory coupled to the bus consists of version mapping mechanism and version loader. The preferred version of default class is sub-class of default class. INDEPENDENT CLAIMS are also included for the following:

1.method for class version naming and mapping;

2.program for class version naming and mapping

USE - For managing version changes of classes in object oriented computer system.
ADVANTAGE - As version mapping mechanism determines and maps most recent best versions of class, an improved version is loaded. An version mapping mechanism maps to recent version, user is able to create class without any knowledge of actual class version .
DESCRIPTION OF DRAWINGS - The figure shows the flowchart which shows the process involved in loading class.

Title Terms/Index Terms/Additional Words: CLASS; VERSION; MAP ; SYSTEM; OBJECT; ORIENT; COMPUTER

Class Codes
International Classification (Main): G06F-017/30

File Segment: EPI;

DWPI Class: T01
Manual Codes (EPI/S-X): T01-J05B

Class version naming and mapping system in object oriented computer system

Original Titles:
Method and apparatus for class version naming and mapping .

Alerting Abstract ...request is made to load a default class, a version loader object checks a version mapping mechanism to determine a preferred version of default class. The preferred version of default class...
DESCRIPTION - A memory coupled to the bus consists of version mapping mechanism and version loader. The preferred version of default class is sub-class of default...

...method for class version naming and mapping; program for class version naming and mapping
...

...USE - For managing version changes of classes in object oriented computer system.
...

...ADVANTAGE - As version mapping mechanism determines and maps most recent best versions of class, an improved version is loaded. An version mapping mechanism maps to recent version, user is able to create class without any knowledge of actual class version .

Title Terms.../Index Terms/Additional Words: MAP ;

Original Publication Data by Authority

Original Abstracts:
...a preferred embodiment of the present invention, a class versioning and mapping system allows a user to request a desired class without knowing which class version is the most recent or correct version for the desired class. The class versioning and mapping system uses a version mapping mechanism to cross reference the requested class, select the most recent or best version of the requested class, and...
Claims:
...package name;a version mapping mechanism residing in the memory, the version mapping mechanism including a mapping of classes to preferred class versions and their names;a version loader object residing in the memory;wherein the JAVA package name for each of the...

...mapping mechanism using the version class identifier to determine a preferred version of the requested JAVA class, then loads the preferred version of the requested JAVA class.Basic Derwent Week: 199952

24/69,K/42 (Item 34 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0009611507 - Drawing available
WPI ACC NO: 1999-561210/ 199947
XRPX Acc No: N1999-414665
Multi-stage software upgrading method for telephone switching system
Patent Assignee: LUCENT TECHNOLOGIES INC (LUCE)
Inventor: MAO V C; UY A T
Patent Family (1 patents, 1 countries)
Patent Application

Number	Kind	Date	Number	Kind	Date	Update
US 5960205	A	19990928	US 1997960642	A	19971030	199947 B

Priority Applications (no., kind, date): US 1997960642 A 19971030

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
US 5960205	A	EN	6	3	

Alerting Abstract US A

NOVELTY - The program text and office dependent data are inserted into files of two duplicate disk file controllers (11,12) which are correspondingly connected to the two duplicate processors (15,16). The duplicate processors are connected to peripherals of switching system to perform switching control.

DESCRIPTION - The office dependent data is derived by iteratively deriving intermediate release office dependent data from a previous release of office dependent data.

USE - For controlling telephone switching system or other real time system.

ADVANTAGE - Arrangement avoids necessity for additional software development for conversion of database for a jump in releases and also avoids long process of implementing multiple release change through series of single release changes.

DESCRIPTION OF DRAWINGS - The figure shows the configuration of a system being upgraded through multiple releases of the program and through a change of processors and disk file controls.

11,12 Duplicate disk file controllers

15,16 Duplicate processors

Title Terms/Index Terms/Additional Words: MULTI; STAGE; SOFTWARE; UPGRADING ; METHOD; TELEPHONE; SWITCH; SYSTEM

Class Codes
International Classification (Main): G06F-009/45

File Segment: EPI;
DWPI Class: T01
Manual Codes (EPI/S-X): T01-F05A

Multi-stage software upgrading method for telephone switching system

Alerting Abstract DESCRIPTION - The office dependent data is derived by iteratively deriving intermediate release office dependent data from a previous release of office dependent data...

...DESCRIPTION OF DRAWINGS - The figure shows the configuration of a system being upgraded through multiple releases of the program and through a change of processors and disk file controls...

Original Publication Data by Authority

Original Abstracts:
...and a target release version of office dependent data. This target release version is derived by upgrading the office dependent data one release at a time until office dependent data for the target release is derived. The new program text and target release version of office dependent data is inserted into one member of a...

...duplicate processors is connected to the disk file controller, if disk file controllers have been changed , and the other disk file is updated with the new program text and office dependent data. Advantageously, this arrangement allows a multiple release jump to be made without

requiring special...
Claims:
In a telephone switching system, a method for performing a multi-stage software upgrade to a target release, comprising the steps of: (a) in said telephone switching system, providing duplicate processors...

...file controllers, each controller comprising a disk file; (b) deriving program text for the target release ; (c) deriving office dependent data for the target release by iteratively deriving intermediate release office dependent data , in each case from a previous release of office dependent data; (d) inserting said program text and said office dependent...

24/69,K/47 (Item 39 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0009455647 - Drawing available
WPI ACC NO: 1999-395493/ 199933
Related WPI Acc No: 1999-395494; 1999-395495; 1999-395496; 1999-395497; 1999-395498; 1999-395500; 1999-395502; 1999-405226
XRPX Acc No: N1999-295593
Data processing system
Patent Assignee: BAAN DEV BV (BAAN-N); LEARMONT T R (LEAR-I); NG T C T (NGTC-I); SUN MICROSYSTEMS INC (SUNM)
Inventor: LEARMONT T R; NG T C T; SUN MICROSYSTEMS I; TUNG NG T C
Patent Family (10 patents, 81 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update	
WO 1999032996	A1	19990701	WO 1998US27187	A	19981221	199933	B
AU 199922024	A	19990712	AU 199922024	A	19981221	199950	E
EP 1042721	A1	20001011	EP 1998966033	A	19981221	200052	E
			WO 1998US27187	A	19981221		
US 6268850	B1	20010731	US 199768415	P	19971222	200146	E
			US 1998106212	A	19980629		
US 20010018690	A1	20010830	US 199768415	P	19971222	200151	E
			US 1998105957	A	19980629		
			US 2001754263	A	20010105		
US 6385618	B1	20020507	US 199768415	P	19971222	200235	E
			US 1998105957	A	19980629		
JP 2003517650	W	20030527	WO 1998US27187	A	19981221	200344	E
			JP 2000525832	A	19981221		
US 6609133	B2	20030819	US 199768415	P	19971222	200356	E
			US 1998105957	A	19980629		
			US 2001754263	A	20010105		
EP 1042721	B1	20040421	EP 1998966033	A	19981221	200428	E
			WO 1998US27187	A	19981221		
DE 69823388	E	20040527	DE 69823388	A	19981221	200436	E
			EP 1998966033	A	19981221		
			WO 1998US27187	A	19981221		

Priority Applications (no., kind, date): US 2001754263 A 20010105; US 1998106212 A 19980629; US 199768415 P 19971222; US 1998105957 A 19980629

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
WO 1999032996	A1	EN	41	12	

National Designated States,Original: AL AM AT AU AZ BA BB BG BR BY CA CH CN CU CZ DE DK EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT UA UG UZ VN YU ZW
Regional Designated States,Original: AT BE CH CY DE DK EA ES FI FR GB GH

GM GR IE IT KE LS LU MC MW NL OA PT SD SE SZ UG ZW

AU 199922024	A	EN		Based on OPI patent WO 1999032996
EP 1042721	A1	EN		PCT Application WO 1998US27187
				Based on OPI patent WO 1999032996
Regional Designated States,Original: DE FR GB				
US 6268850	B1	EN		Related to Provisional US 199768415
US 20010018690	A1	EN		Related to Provisional US 199768415
				Continuation of application US 1998105957
US 6385618	B1	EN		Related to Provisional US 199768415
JP 2003517650	W	JA	48	PCT Application WO 1998US27187
				Based on OPI patent WO 1999032996
US 6609133	B2	EN		Related to Provisional US 199768415
				Continuation of application US 1998105957
				Continuation of patent US 6385618
EP 1042721	B1	EN		PCT Application WO 1998US27187
				Based on OPI patent WO 1999032996
Regional Designated States,Original: DE FR GB				
DE 69823388	E	DE		Application EP 1998966033
				PCT Application WO 1998US27187
				Based on OPI patent EP 1042721
				Based on OPI patent WO 1999032996

Alerting Abstract WO A1

NOVELTY - Method is for a computer system database. Database modifications to the database schema are received, object model modifications are received into the object model and a new version of the data structure is created reflecting the schema of the database with the received database modifications . The new version of the data structure is compared with the preexisting version of the data structure to isolate the database modifications and the object model is updated to reflect the database modifications so that the updated object model contains both the database modifications and the object model modifications to generate the source code.

DESCRIPTION - There are INDEPENDENT CLAIMS for (1) a data processing system and (2) a computer system.

USE - System is for integrating both modifications to an object model and modifications to a database into source code by an object-relational mapping tool.

ADVANTAGE - System improves object-relational mapping tools.

DESCRIPTION OF DRAWINGS - The drawing shows a flowchart of the states performed by the object-relational mapping tool.

Title Terms/Index Terms/Additional Words: DATA; PROCESS; SYSTEM

Class Codes

International Classification (Main): G06F-012/00, G06F-013/00, G06F-017/30
(Additional/Secondary): G06F-009/44

File Segment: EPI;
DWPI Class: T01
Manual Codes (EPI/S-X): T01-F05; T01-H05B1; T01-J05B; T01-J05B4M

Original Titles:

...INTEGRATING BOTH MODIFICATIONS TO AN OBJECT MODEL AND MODIFICATIONS TO A DATABASE INTO SOURCE CODE BY AN OBJECT-RELATIONAL MAPPING TOOL...

...INTEGRATING BOTH MODIFICATIONS TO AN OBJECT MODEL AND MODIFICATIONS TO A DATABASE INTO SOURCE CODE BY AN OBJECT-RELATIONAL MAPPING TOOL...

...Integrating both modifications to an object model and modifications to a.databse into source code by an object-relational mapping tool...

...Integrating both modifications to an object model and modifications

to a database into source code by an object-relational mapping tool...

...Integrating both modifications to an object model and modifications to a database into source code by an object-relational mapping tool...

...INTEGRATING BOTH MODIFICATIONS TO AN OBJECT MODEL AND MODIFICATIONS TO A DATABASE INTO SOURCE CODE BY AN OBJECT-RELATIONAL MAPPING TOOL...

Alerting Abstract ...NOVELTY - Method is for a computer system database. Database modifications to the database schema are received, object model modifications are received into the object model and a new version of the data structure is created reflecting the schema of the database with the received database modifications . The new version of the data structure is compared with the preexisting version of the data structure to isolate the database modifications and the object model is updated to reflect the database modifications so that the updated object model contains both the database modifications and the object model modifications to generate the source code....USE - System is for integrating both modifications to an object model and modifications to a database into source code by an object-relational mapping tool...

...ADVANTAGE - System improves object-relational mapping tools...

...OF DRAWINGS - The drawing shows a flowchart of the states performed by the object-relational mapping tool.

Original Publication Data by Authority

Original Abstracts:
...and systems consistent with the present invention, an improved object-relational mapping tool is provided that generates source code containing classes which preserve both changes to the database schema as well as customizations to a preexisting version of the classes. This functionality alleviates the programmer from having to...

...present invention, an improved object-relational mapping tool is provided that generates source code containing classes which preserve both changes to the database schema as well as customizations to a preexisting version of the classes. This functionality alleviates the programmer from having to recreate their changes to the...

...the classes, and an optional mapping tool maps the defined lock groups when converting data between an object model and a relational model...

...invention, an improved object-relational mapping tool is provided that generates source code containing classes which preserve both changes to the database schema as well as customizations to a preexisting version of the classes. This functionality alleviates the programmer from having to recreate their changes to the classes...
...relational mapping tool is provided that generates source code containing classes which preserve both changes to the database schema as well as customizations to a preexisting version of the classes. This functionality alleviates the programmer from having to recreate their changes to the classes when the database changes...

...that generates source code containing classes which preserve both changes to the database schema as well as customizations to a preexisting version of the classes. This functionality alleviates the programmer from having to recreate their changes to the classes when the database changes, thus saving significant development time...

...que les adaptations a une version preexistante des classes. Cette fonctionnalite soulage le programmeur qui n 'a pas a recreer les modifications sur les classes lorsque la base de donnees change, ce qui

permet...
Claims:
...structure reflecting a modified version of the logical structural relationship among the data in the data source (118);</br> comparing the first data structure with the second data structure to isolate the modifications made to the logical structural relationship among the data; and</br> generating source code (120) reflecting the modifications.

...

...the received database modifications;comparing the new version of the data structure with the preexisting version of the data structure to isolate the database modifications; updating the object model to reflect the database modifications such that the updated object model contains both the database modifications and the object model modifications; andgenerating source...

...based on the updated object model.

An apparatus, comprising : a memory having program instructions for permitting a user to view lock groups reflecting a class
...
...database modifications such that the updated object model contains both the database modifications and the object model modifications; and generating source code based on the updated object model.
Basic Derwent Week: 199933

? t24/69,k/52,61,67,74-75

24/69,K/52 (Item 44 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0009289856 - Drawing available
WPI ACC NO: 1999-219873/ 199919
XRPX Acc No: N1999-162711
Program specific image data recursive processing related to factory production lines - executes image processing through command structure served by separate program or parameter edit or storage modules and recalls developed versions from intermediate memory as needed
Patent Assignee: MATSUSHITA DENKI SANGYO KK (MATU)
Inventor: OKUMURA K; SUZUKI N; YOKOYAMA H
Patent Family (1 patents, 1 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update
JP 11053568	A	19990226	JP 1997212876	A	19970807	199919 B

Priority Applications (no., kind, date): JP 1997212876 A 19970807

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
JP 11053568	A	JA	7	4	

Alerting Abstract JP A
NOVELTY - Data storage module (e) with image input (p) and image processor (q) utilizes control/command structure (k) from a memory (f). The memory has program/parameter development submodules (g,h) coupled to their counterparts (a,c,b,d) in the core program /parameter editing/storage modules. Parameter modifier (m) is attached to the parameter development submodule (h). DETAILED DESCRIPTION - Image-data processing through recursion can be initiated from the developed versions in the intermediate memory, references being made to the core program/parameter storage only when necessary.
USE - It finds use in intra-factory goods movement monitoring.
ADVANTAGE - It speeds up data processing by minimizing the frequency of ab initio parametric development in recursive programs. DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of the image-data processor. (a,c,b,d) Core program/parameter editing/storage submodules; (e) Data storage module; (f) Intermediate memory; (g,h) Program/parameter development submodules; (k) Control/command structure; (m) Parameter modifier; (p) Image input; (q) Image processor.

Title Terms/Index Terms/Additional Words: PROGRAM; SPECIFIC; IMAGE; DATA; RECURSIVE; PROCESS; RELATED; FACTORY; PRODUCE; LINE; EXECUTE; THROUGH; COMMAND; STRUCTURE; SERVE; SEPARATE; PARAMETER; EDIT; STORAGE; MODULE; RECALL; DEVELOP; VERSION; INTERMEDIATE ; MEMORY; NEED

Class Codes
International Classification (Main): G06T-011/80
(Additional/Secondary): G06T-007/00

File Segment: EPI;
DWPI Class: T01
Manual Codes (EPI/S-X): T01-F06; T01-J05A2; T01-J07; T01-J10B2; T01-J10C

...through command structure served by separate program or parameter edit or storage modules and recalls developed versions from intermediate memory as needed

Alerting Abstract ...development submodules (g,h) coupled to their counterparts (a,c,b,d) in the core program /parameter editing/storage

modules. Parameter modifier (m) is attached to the parameter development submodule (h). DETAILED DESCRIPTION - Image-data processing through recursion can be initiated from the developed versions in the intermediate memory, references being made to the core program/parameter storage only when necessary...

...ADVANTAGE - It speeds up data processing by minimizing the frequency of ab initio parametric development in recursive programs. DESCRIPTION OF ...

...a,c,b,d) Core program/parameter editing/storage submodules; (e) Data storage module; (f) Intermediate memory; (g,h) Program/parameter development submodules; (k) Control/command structure; (m) Parameter modifier; (p...

Title Terms.../Index Terms/Additional Words: INTERMEDIATE ;

...
...

24/69,K/61 (Item 53 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0008553127 - Drawing available
WPI ACC NO: 1998-086498/ 199808
XRPX Acc No: N1998-068768
Zymbolic history management system for creating annotated symbolic links to files in shared file system - creating files under directory hierarchies of user file system each file being symbolic link corresp to file version comprising shared file system, based on mappings of directory hierarchies
Patent Assignee: HEWLETT-PACKARD CO (HEWP)
Inventor: BURGOON D A
Patent Family (1 patents, 1 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update
US 5706510	A	19980106	US 1996617989	A	19960315	199808 B

Priority Applications (no., kind, date): US 1996617989 A 19960315

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
US 5706510	A	EN	31	7	

Alerting Abstract US A

The zymbolic history management system (ZHMS) allows a user to create annotated symbolic links or zymlinks to files in a shared file system. A single link is created to each file of the shared file system which a user wishes to access. A zymlink to the shared file system is updated via a zupdate command. Rather than selectively updating individual files , a user simply zupdate's whole trees of files. When it comes time to edit files, a user makes use of standard history managed file system commands to obtain a writable copy of a file.

Use of one of these commands destroys a zymlink corresponding to the file, and substitutes a privately-owned copy of the file on a user's disk. When the edited file is checked back into the history managed file system, a read-only copy of the file remains on a user's disk. When a user updates his or her file system, the read-only file copy may be replaced with a new zymlink. A user will normally view current versions of shared files. Rather than viewing current versions of shared files, a user may wish to maintain a static view of file versions by issuing a zreset command. The system provides a user with more free disk space than in conventional history managed systems.

ADVANTAGE - User's file storage remains logically private, but has access to shared files . Designer can undo periodic update to shared file system w.r.t. logically separate view of shared files. Designer has freedom to choose whether he reads shared files or personally owned physical files.

Title Terms/Index Terms/Additional Words: HISTORY; MANAGEMENT; SYSTEM; SYMBOL; LINK; FILE; SHARE; DIRECTORY; USER; CORRESPOND; VERSION; COMPRISE ; BASED

Class Codes
International Classification (Main): G06F-017/30

File Segment: EPI;
DWPI Class: T01
Manual Codes (EPI/S-X): T01-F05E

...each file being symbolic link corresp to file version comprising shared file system, based on mappings of directory hierarchies

Alerting Abstract ...the shared file system which a user wishes to access. A zymlink to the shared file system is updated via a zupdate command. Rather than selectively updating individual files , a user simply zupdate's whole trees of files. When it comes time to edit...

...read-only copy of the file remains on a user's disk. When a user updates his or her file system, the read-only file copy may be replaced with a new zymlink. A user...

...ADVANTAGE - User's file storage remains logically private, but has access to shared files . Designer can undo periodic update to shared file system w.r.t. logically separate view of shared files. Designer has freedom to choose...

Original Publication Data by Authority

Original Abstracts:
...which a user wishes to access. A zymlink to the shared file system may be updated via a zupdate command . Rather than selectively updateing individual files, a user simply zupdate 's whole trees of files. When it comes time to edit files, a user makes...

...the file remains on a user's disk. When a user updates his or her file system, the read- only file copy may be replaced with a new zymlink. A user will normally view current...
Claims:
...specific versions of the plurality of files, the method comprising:a) constructing a plurality of file versions to be stored within a shared file system, said file versions being constructed from a plurality of file revision archives stored in a history managed file system, said shared and history managed file systems being logically separate from...

...to the history managed file system;c) mapping directory hierarchies of the plurality of user file systems into directory hierarchies of the shared file system;d) creating a plurality of files...

...versions being based on the mappings of directory hierarchies;e) creating first revision information files, said first revision information files being derived from said file revision archives , each first revision information file comprising file revision information corresponding to the file versions of one or more directories of the shared file system' s directory hierarchies; andf) creating second revision information files, said second revision information files being initially derived from said first revision

information files, each second revision information file comprising file revision information corresponding to the symbolic links of one or more of a user file system's directory hierarchies.
Basic Derwent Week: 199808

24/69,K/67 (Item 59 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0007976501 - Drawing available
WPI ACC NO: 1997-067593/ 199707
XRPX Acc No: N1997-055591
Method providing versioning information in software program - gives software program object code and mapfile giving version name associated with program version, links code so add is made to code based on mapfile, of information identifying program version name program yielding versioned object
Patent Assignee: SUN MICROSYSTEMS INC (SUNM)
Inventor: EVANS R I; GINGELL R A
Patent Family (5 patents, 6 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update	
EP 752647	A1	19970108	EP 1996110383	A	19960627	199707	B
JP 9152961	A	19970610	JP 1996195380	A	19960705	199733	E
US 5805899	A	19980908	US 1995499062	A	19950706	199843	E
EP 752647	B1	20020529	EP 1996110383	A	19960627	200236	E
DE 69621381	E	20020704	DE 69621381	A	19960627	200251	E
			EP 1996110383	A	19960627		

Priority Applications (no., kind, date): EP 1996110383 A 19960627; US 1995499062 A 19950706

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
EP 752647	A1	EN	36	18	
Regional Designated States,Original: DE FR GB IT NL					
JP 9152961	A	JA	22		
EP 752647	B1	EN			
Regional Designated States,Original: DE FR GB IT NL					
DE 69621381	E	DE			Application EP 1996110383 Based on OPI patent EP 752647

Alerting Abstract EP A1

The versioning provision method provides a first object code for a first software program. A mapfile (130) is provided specifying a version name associated with a version of the first software program.

The first object code is linked so that an addition is made to the first object code, in accordance with the mapfile, of information identifying the version name of the version of the first software program, yielding a versioned object.

USE/ADVANTAGE - For providing dynamic linking to manage changes in successive versions of software programs .

USE/ADVANTAGE - Keeps track of changes in successive versions of objects accessed by application software programs, allowing for controlled evolution of objects, while maintaining backward compatibility between application programs and objects.

Title Terms/Index Terms/Additional Words: METHOD; INFORMATION; SOFTWARE; PROGRAM; OBJECT; CODE; VERSION; NAME; ASSOCIATE; LINK; SO; ADD; MADE; BASED; IDENTIFY; YIELD

Class Codes
International Classification (Main): G06F-009/06, G06F-009/44, G06F-009/45

(Additional/Secondary): G06F-009/445

File Segment: EPI;
DWPI Class: T01
Manual Codes (EPI/S-X): T01-F05B2; T01-F07

Original Titles:
...Method and apparatus for internal versioning of objects using a map file...

Alerting Abstract ...USE/ADVANTAGE - For providing dynamic linking to manage changes in successive versions of software programs .
...

...USE/ADVANTAGE - Keeps track of changes in successive versions of objects accessed by application software programs, allowing for controlled evolution of objects, while maintaining backward compatibility between application programs and objects.

Original Publication Data by Authority

Original Abstracts:
...is compiled and linked, at build time, the link-editor (124) creates a version definition section and a version symbol section in the object that specify the global symbols defined in various versions of the object. The object can be a shared object (114), a relocatable object (112), or a dynamic executable object (120). When an application software program is linked with the versioned object, at build time , the linker-editor (124) creates a version dependency section in the resulting dynamic executable object that specifies which version of the object is required for execution...

...are present before the program is executed.The invention allows the definition of "weak" versions that do not contain new global symbols. The invention also allows the explicit specification of the version of an
...

...at build time, the link-editor creates a version definition section and a version symbol section in the object that specify the global symbols defined in various versions of the object. The object can be a shared object, a relocatable object, or a dynamic executable object. When an application software program is linked with the versioned object, at build time, the linker-editor creates a version dependency section in the resulting dynamic executable object that specifies which version of the object is required for execution of the software application program...

...is executed. The invention allows the definition of "weak" versions that do not contain new global symbols. The invention also allows the explicit specification of the version of an object to which the software...
Claims:
...124) the first object code (112), so that an addition is made to the first object code (112), in accordance with the mapfile (130, 132), of information identifying the version name of the first software program...

.....les informations identifiant les symboles globaux formant l'interface de la version.

...

...first software program, to yield a versioned object, the linking step further including the step of adding information to the first object code, in accordance with the mapfile information, the information identifying the global symbols formingBasic Derwent Week: 199707

24/69,K/74 (Item 66 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0007216860 - Drawing available
WPI ACC NO: 1995-265007/ 199535
XRPX Acc No: N1995-203890
Computer implemented method using pre-existing data structures - modifies pre-existing data -structures to produce shadow-data-structure containing shadows of pre-existing data-structure needed to perform tasks of new version of program
Patent Assignee: SUN MICROSYSTEMS INC (SUNM)
Inventor: DAY M J; GIBBONS J J; GOLDSTEIN T C; JONATHAN; JORDAN M J
Patent Family (6 patents, 5 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update	
EP 665493	A2	19950802	EP 1995300024	A	19950126	199535	B
JP 8036494	A	19960206	JP 199531836	A	19950130	199615	E
EP 665493	A3	19960228	EP 1995300024	A	19950126	199622	E
US 5761511	A	19980602	US 1994187972	A	19940128	199829	E
EP 665493	B1	20000823	EP 1995300024	A	19950126	200041	E
DE 69518446	E	20000928	DE 69518446	A	19950126	200056	E
			EP 1995300024	A	19950126		

Priority Applications (no., kind, date): US 1994187972 A 19940128

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
EP 665493	A2	EN	44	20	
Regional Designated States,Original: DE FR IT NL SE					
JP 8036494	A	JA	31		
EP 665493	A3	EN			
EP 665493	B1	EN			
Regional Designated States,Original: DE FR IT NL SE					
DE 69518446	E	DE			Application EP 1995300024
					Based on OPI patent EP 665493

Alerting Abstract EP A2

The method augments first header files to add a method to each type of object to be shadowed. The method creates a shadow of an object to be shadowed. The first header lines are scanned to look for a declaration of a standard method. If the standard method is found, a corresponding standard definition of the declaration is generated.

The first header files are collected for later use. The first code files are compiled together with the first header files. They produce a first object code corresponding to the first code files and the header files. The first header files collected for later use are combined with the first object code. The combination represents a mechanism to process a shadowable data structure.

USE/ADVANTAGE - Provides type-safe framework for dynamically extensible objects for making large data structure of objects easily accessible to tool developers, applicable to any environment for modifying collection of connected objects of variety of types. Shortens time and cost to create new version of program.

Title Terms/Index Terms/Additional Words: COMPUTER; IMPLEMENT; METHOD; PRE-EXISTING; DATA; STRUCTURE; MODIFIED; PRODUCE; SHADOW; CONTAIN; NEED; PERFORMANCE; TASK; NEW; VERSION; PROGRAM

Class Codes
International Classification (Main): G06F-009/44, G06F-009/45
(Additional/Secondary): G06F-017/30

File Segment: EPI;
DWPI Class: T01
Manual Codes (EPI/S-X): T01-F05A; T01-F07; T01-S

... modifies pre-existing data -structures to produce shadow-data-structure containing shadows of pre-existing data-structure needed to

Alerting Abstract ...large data structure of objects easily accessible to tool developers, applicable to any environment for modifying collection of connected objects of variety of types. Shortens time and cost to create new version of program.

Original Publication Data by Authority

Original Abstracts:
...has the advantages of shortening the time and cost required to create a new version of the computer program . The pre-existing data-structure is modified to produce a shadow -data-structure which contains only shadows of those elements or nodes of the pre-existing data-structure required to perform...

...of the computer program along with the shadow data-structure to create the desired execution. This new version of the computer program is typically a tool for checking or observing the original program's execution in some...
...system and processes disclosed provide mechanisms for a software manufacturer to create type-safe versions of a connected collection of objects which are dynamically extensible...

...the time and cost required to create a new version of the computer program. The pre -existing data- structure is modified to produce a shadow data- structure which contains only shadows of those elements or nodes of the pre-existing data-structure required to perform the tasks of the new...

...with the shadow data-structure to create the desired execution. This new version of the computer program is typically a tool for checking or observing the original program's execution in some manner. Moreover, the system and...

...mechanisms for a software manufacturer to create type-safe versions of a connected collection of objects which are dynamically extensible.
Claims:
...data-structure (183) that corresponds to the first node and to create and store a plurality of pointer mechanisms; andusing said shadow map to relate the shadow node to the...
...the plurality of pointer mechanisms, wherein a first one of the pointer mechanisms points to the first node in said first data-structure (181) and a second one of the pointer...

...node, and if said entry which points to said shadow node does not exist, said first method will create a new shadow node...relate the shadow node to the first node through the use of the plurality of pointer mechanisms, wherein a first one of the pointer mechanisms points to the first node in
...

...a second one of the pointer mechanisms points to the shadow node in said shadow data -structure wherein a first method operative to look-up a shadow node in said shadow...
...

24/69,K/75 (Item 67 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2007 The Thomson Corporation. All rts. reserv.

0006995902
WPI ACC NO: 1995-007026/ 199501
Related WPI Acc No: 1995-098883; 1995-255159
XRPX Acc No: N1995-005670
Data storage appts with data stored in data structure - has number of layers containing view of blob, with portion of data, and blob identifiers
Patent Assignee: APPLE COMPUTER INC (APPY)
Inventor: AUSTIN D J; CELIK T I; HARRIS J; LO S; LO S W; SZYMANSKI S J
Patent Family (9 patents, 45 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update	
WO 1994027232	A1	19941124	WO 1994US5346	A	19940512	199501	B
AU 199468329	A	19941212	AU 199468329	A	19940512	199521	E
EP 698243	A1	19960228	EP 1994916764	A	19940512	199613	E
			WO 1994US5346	A	19940512		
JP 8510347	W	19961029	JP 1994525725	A	19940512	199705	E
			WO 1994US5346	A	19940512		
US 5758347	A	19980526	US 199360809	A	19930512	199828	E
			US 1995470619	A	19950606		
US 5857207	A	19990105	US 199360809	A	19930512	199909	E
			US 199863846	A	19980421		
US 5870764	A	19990209	US 199360809	A	19930512	199913	E
EP 698243	B1	19990728	EP 1994916764	A	19940512	199934	E
			WO 1994US5346	A	19940512		
DE 69419749	E	19990902	DE 69419749	A	19940512	199942	E
			EP 1994916764	A	19940512		
			WO 1994US5346	A	19940512		

Priority Applications (no., kind, date): US 199863846 A 19980421; US 1995470619 A 19950606; US 199360809 A 19930512

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
WO 1994027232	A1	EN	165	30	
National Designated States,Original: AT AU BB BG BR BY CA CH CN CZ DE DK ES FI GB HU JP KP KR KZ LK LU LV MG MN MW NL NO NZ PL PT RO RU SD SE SK UA UZ VN					
Regional Designated States,Original: AT BE CH DE DK ES FR GB GR IE IT LU MC NL OA PT SE					
AU 199468329	A	EN			Based on OPI patent WO 1994027232
EP 698243	A1	EN	165	30	PCT Application WO 1994US5346
					Based on OPI patent WO 1994027232
Regional Designated States,Original: DE FR GB IT					
JP 8510347	W	JA	168		PCT Application WO 1994US5346
					Based on OPI patent WO 1994027232
US 5758347	A	EN			Division of application US 199360809
US 5857207	A	EN			Continuation of application US 199360809
EP 698243	B1	EN			PCT Application WO 1994US5346
					Based on OPI patent WO 1994027232
Regional Designated States,Original: DE FR GB IT					
DE 69419749	E	DE			Application EP 1994916764
					PCT Application WO 1994US5346
					Based on OPI patent EP 698243
					Based on OPI patent WO 1994027232

Alerting Abstract WO A1
The appts contains data stored in a data structure. The data structure

comprises a number of blobs and a number of layers. Each blob contains a portion of the data and has a blob identifier associated with each portion.
Each layer has, associated with it, a view of at least one of the blobs. The blob identifiers, of all the blobs in view of each given layer, are mutually unique. The views associated with at least two of the layers include blobs with identical blob identifiers having different portions of data.
USE/ADVANTAGE - Application programs e.g. word processors and database management programs. Enables highly flexible storage of data for wide variety of application programs.

Title Terms/Index Terms/Additional Words: DATA; STORAGE; APPARATUS; STRUCTURE; NUMBER; LAYER; CONTAIN; VIEW; BLOB; PORTION; IDENTIFY

Class Codes
International Classification (Main): G06F-012/00, G06F-015/40, G06F-017/30
(Additional/Secondary): G11B-027/031

File Segment: EPI;
DWPI Class: T01
Manual Codes (EPI/S-X): T01-J05B2

Original Publication Data by Authority

Original Abstracts:
...to another list of properties. Blops are stored in containers, which map to the physical medium on which the data is stored. Containers can associate respective container handlers which are specific...

...Blops are stored in Containers, which map to the physical medium on which the data is stored. Containers can associate respective Container Handlers which are specific to the Container's physical...

...which map to the physical medium on which the data is stored. Containers associate respective Container Handlers which are specific to the Container's physical medium. Related Blops in a Container...medium on which the data is stored. Containers associate respective Container Handlers which are specific to the Container's physical medium. Related Blops in a Container are organized into Pools of...

...is stored. Containers can associate respective container handlers which are specific to the container's physical medium. Related blops in a container can be organized into pools or blops. Each container...
Claims:
...1. A method for updating data stored in a data structure, said data structure including a plurality of parts (e .g. A.v1, B.v1, A.v2) of data, each part containing a portion of...

...the steps of:</br> creating a version tree (L1, L2, L3) in said data structure, said tree having a plurality of intermediate ones of said layer structures (L2, L3) different from a starting one of said layer structures (L1); and</br> creating a reconciliation one of said layer structures ("reconciliation...

...said intermediate layer structures, said reconciliation layer structure having associated therewith a plurality of parts (A .v1, B.v?, C.v?, D.v1) each having a data part identifier (A, B...

...said intermediate layer structures.

...versions of said work and further supporting the association of a name with each of said versions, comprising the steps of:preparing a first version of said work in said data structure;associating a predefined name

with said first version...
...further supporting the association of a name with each of said layers, comprising the steps of :preparing a first version of said work in said data structure, said first version of said work being divided into the blops which are...

...said second layer having a portion of said second version of said work; andcalling a set- layer -name procedure for said second layer with said predefined name and an identification of said first layer, said...
Basic Derwent Week: 199501